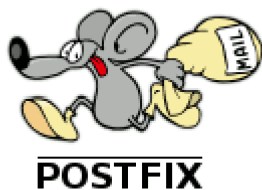
基于 Postfix 邮件系统安装手册

周立军

2011年1月



目　录

序言

从事了多年的网络和Linux系统的管理工作，把自己的经验拿出来和大家一起分享。让大家更多的了解开源软件，使用开源软件。通过本文档可以对邮件系统有个整体的认识。

由于本人水平有限，难免有错误和纰漏之处，请大家多多谅解。

本文档使用OpenOffice 3.2 / LibreOffice 3.3 制作。

免责声明：本文档仅对使用者提供参考，出现问题请使用者自行解决，作者不承担任何责任。

2009年1月--2011年1月

一. 基础知识介绍和基本软件安装与配置

本文中文字描述部分很少，有文字描述部分一般都比较重要，所以请大家仔细看有文件描述的部分。

1.1 E-Mail 系统基础知识

1.1.1 E-Mail 基本概念

E-Mail 系统是一个复杂的系统，涉及到很多复杂的主题，如 SMTP 协议、POP3 协议、IMAP 协议、DNS 协议、垃圾邮件过滤等。

1.1.2 RFC 文档

RFC(Request Comments Document)定义 Internet 的各项标准。关于 E-Mail 的 RFC 文档有好多。下面是想关参考信息，相关 rfc 文档请参阅下面地址：
http://www.ietf.org/iesg/1rfc_index.txt

SMTP 相关 RFC 文档:

http://www.ietf.org/rfc/rfc821.txt
http://www.ietf.org/rfc/rfc2821.txt
http://www.ietf.org/rfc/rfc822.txt
http://www.ietf.org/rfc/rfc2822.txt

POP3 相关 RFC 文档:

http://www.ietf.org/rfc/rfc1939.txt

IMAP 相关 RFC 文档:

http://www.ietf.org/rfc/rfc3501.txt
http://www.ietf.org/rfc/rfc2195.txt
http://www.ietf.org/rfc/rfc2060.txt
http://www.ietf.org/rfc/rfc2221.txt

1.1.2 邮件代理

邮件在传输用会使用 3 种“代理程序”(agent),下面进行逐一介绍。

1. MUA (Mail User Agent)
提供用户写信、读信、寄信、收信的软件。寄信的时候通过 SMTP 协议将邮件交给 MTA ，收信的时候使用 POP3 或 IMAP 协议访问服务器上的用户邮箱。
比较常见的 MUA 软件有 ThunderBird ，FoxMail ，Eudora ，mutt ， Evolution 的 C/S 结构的，也有像 SquirrelMail ，openwebmail，RoundCube ，sqwebmail 等 B/S 结构的 MUA 程序。

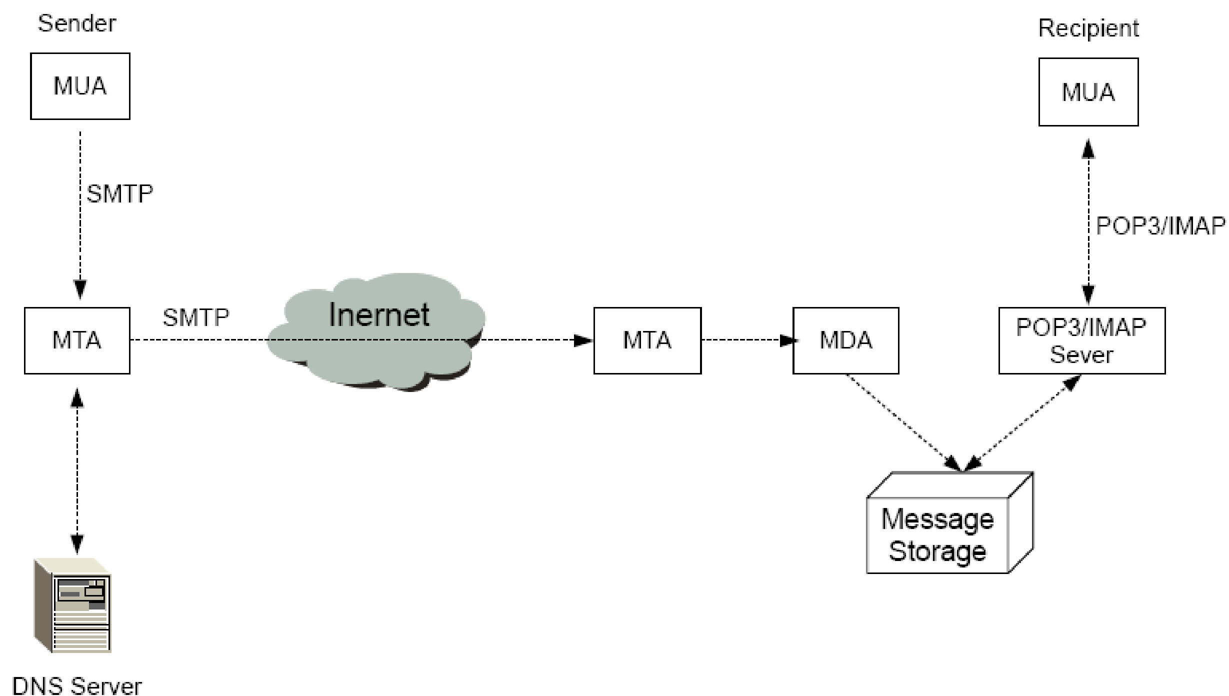
2. MTA (Mail Transfer Agent)
提供接收、传递邮件的服务器软件。决定邮件传递的路径，进行必要的改写地址改写。如果是自己所管辖的域的邮件，就收下邮件，交给 MDA 进行最后的投递。
比较著名的 MTA 有 sendmail ，postfix ，qmail ，extm ，Courier（其中包括 mta，mda 等）等，还有一些非开源的。

3. MDA (Mail Delivery Agent)
MDA 是被 MTA 调用，负责将邮件投递到用户的邮箱。MDA 也可以过滤邮件内容，或是按照用户的规则，

将邮件分类到适当的邮箱；甚至可以将邮件转回给MTA，以寄到另一个邮箱中。

在类Unix的系统中，procmail和maildrop是比较著名的MDA程序，在dovecot软件中也有MDA的功能。

1.1.3 邮件系统结构图



1.2 软件介绍

1.2.1 操作系统

Centos 5

1.2.2 基本软件

Apache
PHP
Mysql
Openssl
Cyrus-Sasl

1.2.3 Mail系统相关软件

Postfix
Dovecot
Postfixadmin
RoundCubemail

1.2.4 病毒扫描与垃圾邮件防护

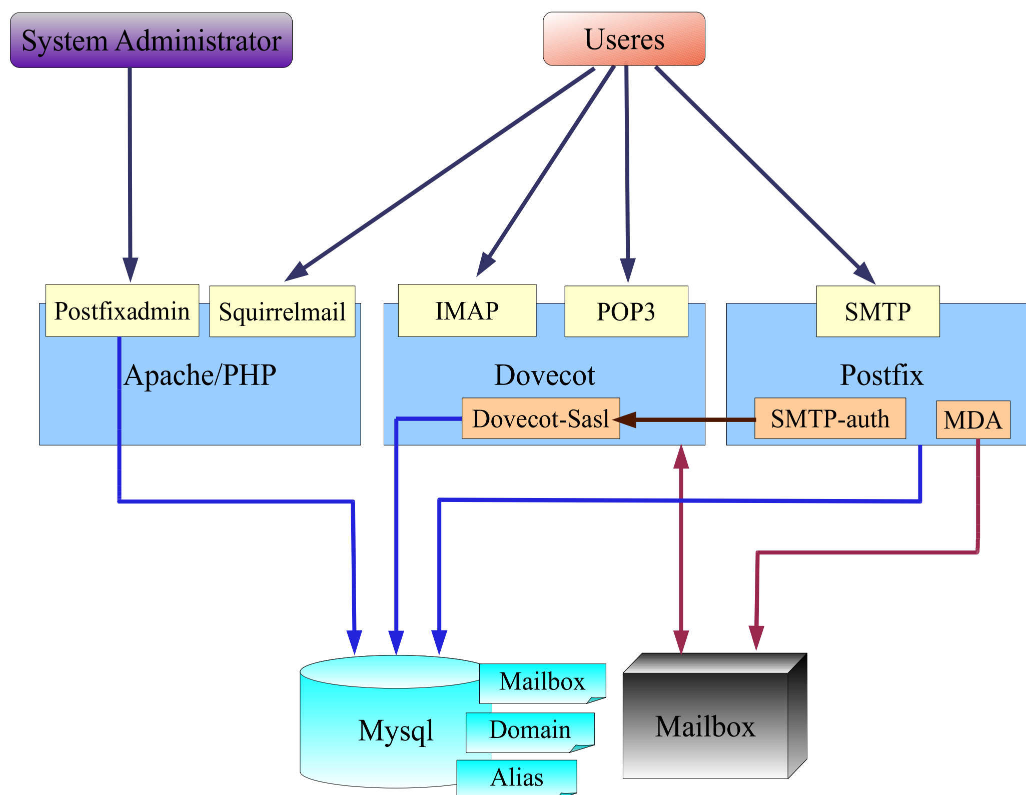
ClamAV
SpamAssassin
MailScanner
MailWatch

1.2.5 日志分析

Mailgraph
Awstats

1.3 postfix 与其他组件之间的关系

1.3.1 基本邮件系统之间的关系



1.4 安装基本系统

1.4.1 CentOS 5.4 安装

1. 安装步骤略。

小提示：对于 Linux 初学者，在安装系统的时候最好进行完全安装，这样后面不会因为缺少某些软件包，出现一些问题，自己还不知道如何解决，造成后面的步骤无法继续。

2. 配置主机名

编辑 /etc/hosts 文件，让文件中只有下面内容：

```
127.0.0.1              localhost.localdomain localhost
192.168.1.70           red.postfix.internal red
```

编辑 /etc/sysconfig/network 文件，修改 HOSTNAME 的值：

```
HOSTNAME=red.postfix.internal
```

1.4.2 更新系统

1. 修改CentOS的yum源**[可选]**

```
shell# cd /etc/yum.repos.d
shell# mv CentOS-Base.repo CentOS-Base.repo.save
shell# wget http://centos.ustc.edu.cn/CentOS-Base.repo.5
shell# mv CentOS-Base.repo.5 CentOS-Base.repo
```

2. 更新系统

```
shell# yum update -y
shell# reboot
```

1.4.3 安装基本软件包

1. 安装编译软件

```
shell# yum install -y gcc gcc-c++ rpm-build autoconf automake
```

2. 安装apache , php , mysql等

```
shell# yum install -y mysql mysql-server mysql-devel
```

```
shell# yum install -y httpd mod_ssl php php-mysql php-mbstring php-imap php-pear-DB
```

启动httpd

```
shell# /etc/init.d/httpd start
```

启动mysqld

```
shell# /etc/init.d/mysqld start
```

3. 验证 apache ，php ，mysql 是否正确

```
shell# echo "<?php phpinfo(); ?>" >/var/www/html/test.php
```

访问：http://yourserver-ip/test.php ， 检查是否支持mysql，imap 功能。

4. 安装mail工具

```
shell# yum install mutt
```

5. 卸载sendmail

```
shell# yum remove sendmail
```

1.4.4 修改mysql root用户密码

这里值得提出的是，有很多人不修改mysql的缺省口令，这样会有很大的安全隐患。修改root用户密码有很多种方法,这里只介绍一种。通过mysqladmin命令修改密码：

```
shell# mysqladmin password "YourPassWord"
```

小提示：如果最好用系统生成的密码，这样安全些。通过openssl passwd 命令生成，然后取8位做为密码。也可以是用 urandom 生成。

```
shell# openssl passwd -stdin < <(echo) | awk '{print substr($0,1,8)}'
```

```
shell# < /dev/urandom tr -dc A-Z-a-z-0-9 | head -c 8
```

1.4.5 关闭没有用的系统服务

1. 关闭不要的系统服务

```
shell# chkconfig pcmcia off
shell# chkconfig atd off
shell# chkconfig cups off
shell# chkconfig isdn off
shell# chkconfig portmap off
shell# chkconfig nfslock off
shell# chkconfig kudzu off
shell# chkconfig rpcidmapd off
```

2. 开启相关系统服务

```
shell# chkconfig httpd on
shell# chkconfig mysqld on
```

好了，准备工作做完了，可以继续了！

1.5 安装方法小结

本例大部分软件都是将其源码制作成RPM包，然后进行安装，这样的好处是在你需要安装多台服务器是将编译好的rpm包文件复制其他机器上就可以批量安装，这样方便管理多台配置相同的服务器。缺点是只适合 CentOS/Redhat/Fedora 系列的发行版，不适合 Debian/Ubuntu 等发行版；但是，有关软件配置 Debian/Ubuntu 系统也可以作为参考。

二. 安装postfix

2.1 编译postfix的RPM包方并安装

2.1.1 下载postfix源码

postfix的官方站：http://www.postfix.org/
postfix的rpm源码包下载地址：http://ftp.wl0.org/official/
当前版本为2.6.6 ，下载最新的版本：

```
shell# wget http://ftp.wl0.org/official/2.6/SRPMS/postfix-2.6.6-1.src.rpm
```

2.1.2 安装postfix源码包

```
shell# rpm -ivh postfix-2.6.6-1.src.rpm
```

注意：如果出现下面警告信息，警告的意思是 sjmudd 用户不存在，文件的所有者设置成root了，可以忽略。后面，安装其他软件的时候也可能有类似问题。

```
warning: postfix-2.6.6-1.src.rpm: Header V3 DSA signature: NOKEY, key ID e9198f3d
   1:postfix                warning: user sjmudd does not exist - using root
warning: group sjmudd does not exist - using root
......
warning: group sjmudd does not exist - using root
########################################### [100%]
```

2.1.3 修改 postfix.spec 文件

修改postfix.spec文件，打开要支持的一些特性。postfix.spec文件位置为：/usr/src/redhat/SPECS/postfix.spec ，修改该文件，让postfix支持你要的一些特性，比如mysql ，ldap等。

注意：with_mysql 选项使用mysql官方发行版，with_mysql_redhat 使用 redhat 自带的版本，with_sasl 值为2表示支持 sasl 版本2。

```
shell# vi /usr/src/redhat/SPECS/postfix.spec
```

修改成下面成如下样子：

```
%define with_cdb          0
%define with_ldap         0
%define with_mysql        0
%define with_mysql_redhat 1
%define with_pcre         0
%define with_pgsql        0
%define with_sasl         2
%define with_spf          0
%define with_dovecot      0
%define with_tls          1
```

```
%define with_tlsfix       2
%define with_vda          0
```

2.1.4 build postfix rpm包

```
shell# cd /usr/src/redhat/SPECS/
shell# rpmbuild -bb postfix.spec
```

注意：如果出现类似下面错误，请安装相应的软件包，然后重新build postfix软件。

```
shell# rpmbuild -bb postfix.spec
error: Failed build dependencies:
        db4-devel is needed by postfix-2.6.6-1.rhel5.i386
        cyrus-sasl-devel is needed by postfix-2.6.6-1.rhel5.i386
```

使用yum进行安装依赖的软件包

```
shell# yum install -y db4-devel cyrus-sasl-devel
```

2.1.5 安装 postfix rpm包

安装postfix

```
shell# cd /usr/src/redhat/RPMS/i386
shell# rpm -ivh postfix-2.6.6-1.rhel5.i386.rpm
```

安装后postfix后自动建立两个组postdrop和postfix ，一个用户postfix 。

2.2 查看postfix支持的特性

2.2.1 检查 postfix 是否正确

postfix 提供了check命令，可以检查当前postfix的配置是否有问题、文件和目录权限是否正确。可以通过运行下面命令：

```
shell# postfix check
```

这个检查工具秉承这“没有消息就是好消息”的优良Unix风格。

2.2.2 查看postfix支持的查询表的类型

```
shell# postconf -m
```

输出结果：

```
btree
cidr
environ
```

```
hash
mysql
nis
proxy
regexp
static
unix
```

2.2.3 检查支持的SASL Server 插件的类型

```
shell# postconf -a
```

输出结果：

```
cyrus
dovecot
```

2.2.4 检查支持的SASL Client 插件的类型

```
shell# postconf -A
```

输出结果：

```
cyrus
```

2.3 简单postfix配置

Postfix的配置文件有两个 main.cf 和 master.df 文件，通过rpm方式安装的postfix，配置文件在 /etc/postfix/ 目录下。如果是编译安装的，在安装的时候路径是由你自己输入的。

2.3.1 main.cf文件

main.cf 文件集中了 postfix 的所有参数。可以通过文本编辑器编辑 main.cf 文件来改变postfix参数的值；也可以通过 postconf 命令来改变 postfix 参数的值，命令格式：

```
shell# postconf -e <parameter>=<value>
```

1. 简化postfix的配置文件

```
shell# postconf -n > /etc/postfix/main.cf.new
shell# mv /etc/postfix/main.cf /etc/postfix/main.cf.old
shell# mv /etc/postfix/main.cf.new /etc/postfix/main.cf
```

2. 配置 Postfix，使用Maildir格式存放用户邮件。

修改Postfix配置 /etc/postfix/main.cf ，增加下面内容:

```
home_mailbox = Maildir/
```

2.3.2 master.cf 文件

Postfix 的所有程序，都是有 master daemon 在需要是才启动的。这些服务程序的运行参数都是在 master.cf 配置文件中定的。一般我不用编辑改文件，除非你要增加一些新的功能。

2.4 启动与停止 postfix

2.4.1 启动 postfix 命令

1. 可以使用 Sys V 风格的脚本启动：

```
shell# /etc/init.d/postfix start
```

2. 也可以通过运行 postfix 命令实现：

```
shell# /usr/sbin/postfix start
```

可以按照下面方法验证 postfix 是否正常启动。

1. 查看/var/log/maillog 日志中是否有错误

2. 检查是否监听 25 端口

```
shell# netstat -antp | grep 25
tcp        0      0 0.0.0.0:25          0.0.0.0:*           LISTEN      3200/master
```

3. 检查是否监听 25 端口

```
shell# telnet localhost 25
Trying 127.0.0.1...
Connected to localhost.localdomain (127.0.0.1).
Escape character is '^]'.
220 red.postfix.internal ESMTP Postfix
ehlo localhost
250-red.postfix.internal
250-PIPELINING
250-SIZE 10240000
250-VRFY
250-ETRN
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
quit
221 2.0.0 Bye
Connection closed by foreign host.
```

2.4.2 停止 postfix 命令

1. 可以使用 Sys V 风格的脚本启动：

```
shell# /etc/init.d/postfix stop
```

2. 也可以通过运行 postfix 命令实现：

```
shell# /usr/sbin/postfix stop
```

2.4.3 重新读取 postfix 配置

1. 可以使用 Sys V 风格的脚本启动：

```
shell# /etc/init.d/postfix reload
```

2. 也可以通过运行 postfix 命令实现：

```
shell# /usr/sbin/postfix reload
```

2.5 常用 postfix 命令

2.5.1 查看 postfix 队列管理工具

postfix 提供了管理队列的工具，可以队列中的邮件进行如下管理：

- 查看邮件队列
- 重新排队
- 保留队列
- 删除队列
- 清空队列
- 擦看邮件内容

1. 查看邮件队列

这个命令和 sendmail 的 mailq 命令一样，能够显示队列中的邮件。

```
shell# postqueue -p
-Queue ID- --Size-- ----Arrival Time---- -Sender/Recipient-------
635AF8E61A      600 Fri Jul 23 12:11:16  root@red.postfix.internal
                                                  (unknown user: "aaaa")
                                         aaaa@red.postfix.internal
```

2. 刷新队列

刷新队列命令可以将队列中的所有邮件重新投递。

```
shell# postqueue -f
```

3. 立即投递队列中指定邮件

如果队列中指定的邮件需要投递，可以使用下面命令。

```
shell# postqueue -i 635AF8E61A
```

4. 保留队列中的邮件

当你想将队列中的邮件无限期的保留在系统中，hold queue 就是这些邮件的家了。

```
shell# postsuper -h 635AF8E61A
```

输出结果：

```
postsuper: 635AF8E61A: placed on hold
postsuper: Placed on hold: 1 message
```

现在已经将改邮件放到 hold 对了中了。

```
shell# ls -l /var/spool/postfix/hold/
total 4
-rwx------ 1 postfix postfix 839 Jul 23  2010 635AF8E61A
```

当你想将 hold queue 中的邮件移回对队列中，可以经 postsuper 命令的选项 -h 换成 -H 即可。

```
shell# postsuper -H 635AF8E61A
```

输出结果：

```
postsuper: 635AF8E61A: released from hold
postsuper: Released from hold: 1 message
```

5. 查看队列中邮件内容

postcat 命令可以查看队列中邮件内容。

```
shell# postcat -q 635AF8E61A
```

6. 重置邮件队列

如果因为配置问题导致邮件中存储了错误信息，以致于邮件不能被发送，在问题解决之后，可能需要将队列中的邮件重新走一遍流程，以便邮件能够发送成功。

重置指定邮件队列：

```
shell# postsuper -r 635AF8E61A
```

输出结果：

```
postsuper: 635AF8E61A: requeued
postsuper: Requeued: 1 message
```

重置所有邮件队列：

```
shell# postsuper -r ALL
```

输出结果：

```
postsuper: Requeued: 5 messages
```

7. 删除队列

删除队列中指定邮件。

```
shell# postsuper -d 254D68E620
```

输出结果：

```
postsuper: 254D68E620: removed
postsuper: Deleted: 1 message
```

8. 删除队列全部

注意:参数 ALL 全部为大写。

```
shell# postsuper -d ALL
```

输出结果：

```
postsuper: Deleted: 3 messages
```

以上命令，如果改变了任何队列信息都会在日志中记录。

2.6 SASL 身份验证

SMTP 服务器可以判断那些 smtp 客户端可以发送邮件，这就需要 SASL 的支持。Postfix 本身没有 SASL 功能，需要第三方软件支持，目前 postfix 支持 Cyrus SASL 和 Dovecot SASL 。

使用 Dovecot SASL 配置起来非常简单，后面在做虚拟用户的时候将使用到 Dovecot SASL，那里有详细的配置。

我在研究 Cyrus SASL 的时候费了不少周折，下面介绍一下 Cyrus SASL 软件的配置。

2.6.1 关于 Cyrus SASL

1. 的认证机制

认证机制由 mech_list 参数指定，下面是详细说明。

- 匿名：anonymous
- 明文文本：PLAIN，LOGIN 。两者通过 base64 编码传输用户名密码，LOGIN 是非便准认证方式，有些特殊的 MUA 客户端使用。
- 加密密码：CRAM-MD5，DIGEST-MD5 。

2. Cyrus SASL 的认证方法

认证方法由 pwcheck_method 参数指定，下面是详细说明。

- saslauthd ：saslauthd 是一个单独的守护进程。可以支持 pam，shadow，sasldb2 等方式。
- auxprop : auxprop 提供辅助的认证方式，可以支持 SQL，LDAP 等。
- authdaemond : authdaemond 是 Courier 的密码认证服务。

3. Cyrus SASL 的日志

log_level 日志可选值为 0～7 ，0 不记录日志，我在试验中没发现这个值有啥用途。

2.6.2 postfix 主要的 SASL 认证参数

- smtpd_sasl_auth_enable : 是否启用 sasl 认证,默认值是 no 。
- smtpd_sasl_type : sasl 认证方式，可以选值 cyrus 或 dovecot ,默认是 cyurs。
- smtpd_sasl_path : smtpd_sasl_path 参数指定 postfix 读取 sasl 的配置文件，不要被 path 给迷惑了，实际上这个值是文件名的前缀。一般 Postfix 会到 /usr/lib/sasl2/ 目录下找配置文件，文件名为“配置值.conf”比如你配置了 smtpd_sasl_path = smtpd ,那么 postfix 会找 /usr/lib/sasl2/smtpd.conf 文件。
- mtpd_sasl_security_options : sasl 认证安全选项，一般用 noanonymous 。
- broken_sasl_auth_clients : 非标准 MUA 支持，默认值 no 。

2.6.4 配置 SASL 认证

这里我们使用 mysql 作为后台数据，配置 sasl ，下面是详细步骤。

1. 安装 Cyrus SASL 软件

一般情况下，系统中都会安装 Cyrus SASL 软件的，好多软件都依赖 Cyrus SASL 软件包。

```
shell# rpm -qa | grep sasl
```

输出结果：

```
cyrus-sasl-lib-2.1.22-5.el5_4.3
cyrus-sasl-plain-2.1.22-5.el5_4.3
cyrus-sasl-2.1.22-5.el5_4.3
cyrus-sasl-sql-2.1.22-5.el5_4.3
cyrus-sasl-devel-2.1.22-5.el5_4.3
cyrus-sasl-md5-2.1.22-5.el5_4.3
```

注意：在 CentOS 系统中 cyrus-sasl-sql 和 cyrus-sasl-md5 没有安装，需要自己手动安装。

```
shell# yum install -y cyrus-sasl-sql cyrus-sasl-md5
```

2. 配置数据

建立 mysql 用户

```
mysql> GRANT ALL PRIVILEGES ON test.* TO 'mytestuser'@'localhost' IDENTIFIED BY 'pa5swd';
mysql> flush privileges;
```

建立表，插入一条记录

```
use test;
CREATE TABLE users (
    username varchar(255) NOT NULL ,
    password varchar(255) NOT NULL default '$$$$$$',
    clear_password varchar(255) NOT NULL default '888888',
    maildir varchar(255) NOT NULL default '',
    mailquota integer NOT NULL DEFAULT '20',
    created TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    modified_by varchar(255) NOT NULL default '',
    active integer NOT NULL default '1',
    PRIMARY KEY (username)
);
insert into users (username,clear_password) values('test@postfixtest.com','12345678');
```

3. 建立 sasl 配置文件

本例的配置文件为：/usr/lib/sasl2/postfix.conf ，内容如下：

```
log_level: 5
pwcheck_method: auxprop
auxprop_plugin: sql
mech_list: CRAM-MD5 PLAIN LOGIN
sql_engine: mysql
sql_hostnames: localhost
sql_user: mytestuser
sql_passwd: pa5swd
sql_database: test
sql_verbose: yes
sql_select: SELECT clear_password FROM users WHERE username='%u@%r' AND active=1
```

4. 配置 postfix 支持 sasl

在 /etc/postfix/main.cf 中加入如下内容：

```
smtpd_sasl_auth_enable      = yes
smtpd_sasl_security_options = noanonymous
smtpd_sasl_path             = postfix
broken_sasl_auth_clients    = yes
smtpd_sasl_authenticated_header = yes
```

配置完成了，重启动 postfix 服务，是配置生效

2.6.5 测试 sasl 认证功能

1. 生成 base64 编码

用户名的 base64 编码：

```
shell# perl -MMIME::Base64 -e "print encode_base64('test@postfixtest.com');"
dGVzdEBwb3N0Zml4dGVzdC5jb20=
```

密码的 base64 编码：

```
shell# perl -MMIME::Base64 -e "print encode_base64('12345678');"
MTIzNDU2Nzg=
```

2. 测试认证发送邮件

```
shell# telnet localhost 25
Trying 127.0.0.1...
Connected to localhost.localdomain (127.0.0.1).
Escape character is '^]'.
220 red.postfix.internal ESMTP Postfix
ehlo localhost
250-red.postfix.internal
250-PIPELINING
250-SIZE 10240000
250-VRFY
250-ETRN
250-AUTH CRAM-MD5 LOGIN PLAIN
250-AUTH=CRAM-MD5 LOGIN PLAIN
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
auth login
334 VXNlcm5hbWU6
dGVzdEBwb3N0Zml4dGVzdC5jb20=
334 UGFzc3dvcmQ6
MTIzNDU2Nzg=
235 2.7.0 Authentication successful
mail from:test@postfixtest.com
250 2.1.0 Ok
rcpt to:zhoulj
250 2.1.5 Ok
data
354 End data with <CR><LF>.<CR><LF>
subject:sasl test mail
from:test@postfixtest.com
to:zhoulj
This is sasl test mail.
.
250 2.0.0 Ok: queued as 52BF59C555
quit
221 2.0.0 Bye
Connection closed by foreign host.
```

3. 查看邮件

```
shell# ls /home/zhoulj/Maildir/new/
```

```
1279858989.V802Ib3b0dM196766.red.postfix.internal
```

4. 查看日志

```
Jul 23 12:21:45 red postfix/smtpd[3137]: connect from localhost.localdomain[127.0.0.1]
Jul 23 12:22:50 red postfix/smtpd[3137]: 52BF59C555:
client=localhost.localdomain[127.0.0.1], sasl_method=login,
sasl_username=test@postfixtest.com
Jul 23 12:23:09 red postfix/cleanup[3161]: 52BF59C555: message-
id=<20100723042250.52BF59C555@red.postfix.internal>
Jul 23 12:23:09 red postfix/qmgr[3064]: 52BF59C555: from=<test@postfixtest.com>, size=435,
nrcpt=1 (queue active)
Jul 23 12:23:09 red postfix/local[3162]: 52BF59C555: to=<zhoulj@red.postfix.internal>,
orig_to=<zhoulj>, relay=local, delay=33, delays=33/0.07/0/0.1, dsn=2.0.0, status=sent
(delivered to maildir)
Jul 23 12:23:09 red postfix/qmgr[3064]: 52BF59C555: removed
Jul 23 12:23:21 red postfix/smtpd[3137]: disconnect from localhost.localdomain[127.0.0.1]
```

5. 查看邮件内容

```
Return-Path: <test@postfixtest.com>
X-Original-To: zhoulj
Delivered-To: zhoulj@red.postfix.internal
Received: from a (localhost.localdomain [127.0.0.1])
        (Authenticated sender: test@postfixtest.com)
        by red.postfix.internal (Postfix) with ESMTPA id 52BF59C555
        for <zhoulj>; Fri, 23 Jul 2010 12:22:36 +0800 (CST)
subject:sasl test mail
from:test@postfixtest.com
to:zhoulj@red.postfix.internal
Message-Id: <20100723042250.52BF59C555@red.postfix.internal>
Date: Fri, 23 Jul 2010 12:22:36 +0800 (CST)

This is sasl test mail.
```

6. 删除 Cyrus SASL 配置

经过以上测试，Postfix 使用 Cyrus SASL 认证的功能已经配置完成了。由于在配置虚拟域的时候我们使用的是 Dovecot SASL 配置，所以我们把 Postfix 中有关 sasl 的配置删除掉。

三. 安装 dovecot

3.1 编译 Dovecot 的 RPM 包方并安装

3.1.1 下载 doevcot 软件包

doevcot 的官方网站：http://www.dovecot.org/
doevcot 源码的 rpm 包下载地址：http://atrpms.net/name/dovecot/
本文档以 1.2 版为例，下载地址：http://packages.atrpms.net/name/dovecot-1.2.x/

```
shell# wget http://dl.atrpms.net/all/dovecot-1.2.12-1_109.src.rpm
```

3.1.2 安装 dovecot 源码

```
shell# rpm -ivh dovecot-1.2.12-1_109.src.rpm
```

3.1.3 修改 dovecot.spec 文件

文件位置：/usr/src/redhat/SPECS/
如果不想让支持某些不需要的功能请修改或者注释相关内容。

如果不需要支持 ldap ，修改 39 行

```
BuildRequires: openldap-devel, cyrus-sasl-devel
```

改成

```
BuildRequires: cyrus-sasl-devel
```

注释 45 行，60 行，62 行，下面内容

```
%{?with_solr:BuildRequires: libcurl-devel expat-devel}
```

```
BuildRequires: postgresql-devel
```

```
BuildRequires: sqlite-devel
```

修改 %configure 部分内容：
删除下面内容：

```
    --with-pgsql                   \
```

```
    --with-sqlite                  \
```

```
    --with-ldap=plugin             \
```

修改下面内容：

```
    --without-bzlib             \
    --without-bzlib               \
    %{?with_solr:--with-solr}
```

改成

```
    --without-bzlib
```

改成

```
    --without-bzlib
```

在

```
%define ssldir %{_sysconfdir}/pki/%{name}
```

下面加入一行

```
%define _initddir /etc/rc.d/init.d
```

3.1.4 建立 dovecot rpm 包

```
shell# cd /usr/src/redhat/SPECS/
shell# rpmbuild -bb dovecot.spec
```

注意：如果出现以下错误，说明系统缺少软件包，请根据提示安装依赖的软件，可以使用 yum 安装，安装完毕后后重新进行 rpmbuild。

```
shell# rpmbuild -bb dovecot.spec
error: Failed build dependencies:
        libtool is needed by dovecot-1.2.12-1_109.i386
        gettext-devel is needed by dovecot-1.2.12-1_109.i386
        pam-devel is needed by dovecot-1.2.12-1_109.i386
        pkgconfig is needed by dovecot-1.2.12-1_109.i386
        bzip2-devel is needed by dovecot-1.2.12-1_109.i386
        libcap-devel is needed by dovecot-1.2.12-1_109.i386
```

通过 yum 安装需要的软件包

```
shell# yum install -y libtool gettext-devel pam-devel pkgconfig bzip2-devel libcap-devel
```

3.1.5 安装 dovecot rpm 包

```
shell# cd /usr/src/redhat/RPMS/i386
shell# rpm -ivh dovecot-1.2.12-1_109.i386.rpm
```

3.2 配置文件 dovecot

Dovecot 的配置文件为 /etc/dovecot.conf ，下面是支持本地用户的配置：

```
protocols = imap pop3
#add by zhoulj for local users at 2009-05-06
mail_location = maildir:~/Maildir
protocol imap {
}
protocol pop3 {
}
protocol lda {
  postmaster_address = postmaster@postfixtest.com
}
auth default {
  mechanisms = plain login
  passdb pam {
  }
  userdb passwd {
  }
  socket listen {
    master {
      path = /var/run/dovecot/auth-master
      mode = 0600
      user = postfix
    }

    client {
      path = /var/spool/postfix/private/auth
      mode = 0660
      user = postfix
      group = postfix
    }
  }
}
dict {
}
plugin {
}
```

3.3 启动与停止 dovecot

3.3.1 启动 dovecot 命令

```
shell# /etc/init.d/dovecot start
```

验证 dovecot 是否正常启动

1. 查看/var/log/maillog 日志中是否有错误

2. 检查是否监听 110 和 143 端口

```
shell# netstat -antp | egrep ":110|:143"
tcp      0       0 0.0.0.0:110      0.0.0.0:*      LISTEN       2910/dovecot
```

```
tcp        0      0 0.0.0.0:143         0.0.0.0:*        LISTEN      2910/dovecot
tcp        0      0 :::110              :::*             LISTEN      2910/dovecot
tcp        0      0 :::143              :::*             LISTEN      2910/dovecot
```

3. 验证 POP3 服务是否正常

```
shell# telnet localhost 110
Trying 127.0.0.1...
Connected to localhost.localdomain (127.0.0.1).
Escape character is '^]'.
+OK Dovecot ready.
quit
+OK Logging out
Connection closed by foreign host.
```

4. 验证 IMP4 服务是否正常

```
shell# telnet localhost 143
Trying 127.0.0.1...
Connected to localhost.localdomain (127.0.0.1).
Escape character is '^]'.
* OK Dovecot ready.
a logout
* BYE Logging out
a OK Logout completed.
Connection closed by foreign host.
```

3.3.2 停止 dovecot 命令

```
shell# /etc/init.d/dovecot stop
```

3.4.3 重新读取 postfix 配置

```
shell# /etc/init.d/postfix reload
```

3.4 Dovecot 日志设置

在默认不更改设置情况下，在 CentOS 系统中 Dovecot 将日志记录到 /var/log/maillog 中，Debian 系统中 Dovecot 将日志记录到 /var/log/mail.log 中。

3.4.1 修改日志路径

```
log_path = /var/log/dovecot.log
info_log_path = /var/log/dovecot-info.log
```

3.4.2 轮询日志设置

修改 /etc/logrotate.d/dovecot 文件

```
# dovecot SIGUSR1: Re-opens the log files.
/var/log/dovecot*.log {
  missingok
  notifempty
  delaycompress
  sharedscripts
  postrotate
    /bin/kill -USR1 `cat /var/run/dovecot/master.pid 2>/dev/null` 2> /dev/null || true
  endscript
}
```

3.4.3 日志级别设置

1. 用户登录认证相关日志

```
auth_verbose=yes
auth_debug=yes
auth_debug_passwords=yes
mail_debug=yes
```

2. SSL 日志

```
verbose_ssl=yes
```

3.5 使用本地用户测试收发邮件

3.5.1 建立一个本地用户

```
shell# useradd localuser
shell# (echo pa5swd;echo pa5swd) | passwd localuser
```

测试 dovecot 是否配置是否正确

```
shell# /usr/libexec/dovecot/authtest localuser
```

下面是正确的输出结果：

```
userdb: localuser
uid   : 501
gid   : 501
user  : localuser
home  : /home/localuser
extra fields:
  system_groups_user=localuser
```

3.5.2 使用 sendmail 发送邮件

```
shell# echo hello | sendmail localuser
```

查看 /var/log/maillog 日志

```
Jul 23 13:09:14 red postfix/pickup[3580]: B7A3E9CC9D: uid=0 from=<root>
Jul 23 13:09:14 red postfix/cleanup[4063]: B7A3E9CC9D: message-
id=<20100723050914.B7A3E9CC9D@red.postfix.internal>
Jul 23 13:09:14 red postfix/qmgr[3579]: B7A3E9CC9D: from=<root@red.postfix.internal>,
size=301, nrcpt=1 (queue active)
Jul 23 13:09:14 red postfix/local[4065]: B7A3E9CC9D: to=<localuser@red.postfix.internal>,
orig_to=<localuser>, relay=local, delay=0.13, delays=0.11/0.02/0/0.01, dsn=2.0.0,
status=sent (delivered to maildir)
Jul 23 13:09:14 red postfix/qmgr[3579]: B7A3E9CC9D: removed
```

3.5.3 使用 SMTP 发送邮件

```
shell# telnet localhost 25
Trying 127.0.0.1...
Connected to localhost.localdomain (127.0.0.1).
Escape character is '^]'.
220 red.postfix.internal ESMTP Postfix
ehlo localhost
250-red.postfix.internal
250-PIPELINING
250-SIZE 10240000
250-VRFY
250-ETRN
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
mail from:root
250 2.1.0 Ok
rcpt to:localuser
250 2.1.5 Ok
data
354 End data with <CR><LF>.<CR><LF>
from:root
to:localuser
subject:test

This is a test mail from cli.
.
250 2.0.0 Ok: queued as A27859C7E6
quit
221 2.0.0 Bye
Connection closed by foreign host.
```

查看 /var/log/maillog

```
Jul 23 13:16:36 red postfix/smtpd[4080]: connect from localhost.localdomain[127.0.0.1]
Jul 23 13:17:44 red postfix/smtpd[4080]: A27859C7E6:
client=localhost.localdomain[127.0.0.1]
Jul 23 13:19:04 red postfix/cleanup[4083]: A27859C7E6: message-
```

```
id=<20100723051744.A27859C7E6@red.postfix.internal>
Jul 23 13:19:04 red postfix/qmgr[3579]: A27859C7E6: from=<root@red.postfix.internal>,
size=403, nrcpt=1 (queue active)
Jul 23 13:19:05 red postfix/local[4086]: A27859C7E6: to=<localuser@red.postfix.internal>,
orig_to=<localuser>, relay=local, delay=124, delays=124/0.03/0/0.01, dsn=2.0.0,
status=sent (delivered to maildir)
Jul 23 13:19:05 red postfix/qmgr[3579]: A27859C7E6: removed
Jul 23 13:19:07 red postfix/smtpd[4080]: disconnect from localhost.localdomain[127.0.0.1]
```

3.5.4 收邮件测试

1. 查看本地目录

```
shell# cd /home/localuser/Maildir/new
shell# ls -1
1279861754.V802Ib3b11M803516.red.postfix.internal
1279862344.V802Ib3b15M9374.red.postfix.internal
```

2. 测试 POP3 服务

```
shell# telnet localhost 110
Trying 127.0.0.1...
Connected to localhost.localdomain (127.0.0.1).
Escape character is '^]'.
+OK Dovecot ready.
user localuser
+OK
pass pa5swd
+OK Logged in.
list
+OK 2 messages:
1 415
2 517
.
retr 1
+OK 415 octets
Return-Path: <root@red.postfix.internal>
X-Original-To: localuser
Delivered-To: localuser@red.postfix.internal
Received: by red.postfix.internal (Postfix, from userid 0)
        id B7A3E9CC9D; Fri, 23 Jul 2010 13:09:14 +0800 (CST)
Message-Id: <20100723050914.B7A3E9CC9D@red.postfix.internal>
Date: Fri, 23 Jul 2010 13:09:14 +0800 (CST)
From: root@red.postfix.internal (root)
To: undisclosed-recipients:;

hello
.
retr 2
+OK 517 octets
Return-Path: <root@red.postfix.internal>
```

```
X-Original-To: localuser
Delivered-To: localuser@red.postfix.internal
Received: from localhost (localhost.localdomain [127.0.0.1])
        by red.postfix.internal (Postfix) with ESMTP id A27859C7E6
        for <localuser>; Fri, 23 Jul 2010 13:17:01 +0800 (CST)
from: root@red.postfix.internal
to: localuser@red.postfix.internal
subject:test
Message-Id: <20100723051744.A27859C7E6@red.postfix.internal>
Date: Fri, 23 Jul 2010 13:17:01 +0800 (CST)

This is a test mail from cli.
.
quit
+OK Logging out.
Connection closed by foreign host.
```

3. 删除 localuser

localuser 测试完毕后将其删掉。

```
shell# userdel -r localuser
```

3.6 安装 dovecot-sieve 和 dovecot-managesieve

Sieve 是一个筛选 Email 的可编程语言，源于 CMU Cyrus 项目，是由 Cyrus IMAP server 创建的。官方网站：http://pigeonhole.dovecot.org/

后面在 WebMail 部分要给用户提供 Vacation 功能的，需要安装这部份内容；如不需要给用户提供 Vacation 功能，可以跳过这部分内容。

3.6.1 安装 dovecot-devel

sieve 依赖 dovecot-devel 包，所以先安装 dovecot 开发包。

```
shell# cd /usr/src/redhat/RPMS/i386
shell# rpm -ivh dovecot-devel-1.2.12-1_109.i386.rpm
```

3.6.2 安装 dovecot-sieve

rpm 源码包下载地址：http://atrpms.net/name/dovecot-sieve/

1. 安装 dovecot-sieve 源码

```
shell# rpm - ivh dovecot-sieve-0.1.17-5.src.rpm
```

2. 编译 dovecot-sieve rpm 包

```
shell# cd /usr/src/redhat/SPECS
shell# rpmbuild -bb dovecot-sieve.spec
```

3. 安装 dovecot-sieve

```
shell# rpm -ivh /usr/src/redhat/RPMS/i386/dovecot-sieve-0.1.17-5.i386.rpm
shell# rpm -ivh /usr/src/redhat/RPMS/i386/dovecot-sieve-devel-0.1.17-5.i386.rpm
```

3.6.3 安装 dovecot-managesieve

rpm 源码包下载地址：http://atrpms.net/name/dovecot-managesieve/

1. 安装 dovecot-managesieve 源码

```
shell# rpm - ivh dovecot-managesieve-0.11.11-0_4.src.rpm
```

2. 编译 dovecot-managesieve rpm包

```
shell# cd /usr/src/redhat/SPECS
shell# rpmbuild -bb dovecot-managesieve.spec
```

3. 安装 dovecot-managesieve

```
shell# cd /usr/src/redhat/RPMS/i386/
shell# rpm -ivh dovecot-managesieve-0.11.11-0_4.i386.rpm
```

3.6.4 修改 dovecot 配置

在 /etc/dovecot.conf 文件中，添加下面内容：

```
### managesieve setting
protocol managesieve {
  listen = localhost:2000
}
```

在 protocols 后面加入 managesieve ，类似下面内容：

```
protocols = pop3 imap pop3s imaps managesieve
```

在 protocol lda 里面的 mail_plugins 项加入sieve，类似下面内容：

```
protocol lda {
  mail_plugins = quota sieve
  ......
}
```

3.7 小结

经过以上的配置，一个基本的邮件系统已经建立完毕，通过系统用户对帐号进行管理，对于一般的内部测试服务器已经能够满足需求。

四. 虚拟域和虚拟用户配置

4.1 安装 PostfixAdmin

4.1.1 准备工作

1. 添加 vmta 用户

该用户的 uid/gid 要大于 1000

```
shell# groupadd -g 5000 vmta
shell# useradd -g vmta -u 5000 vmta -d /var/vmta -m
shell# usermod -G postfix vmta
```

2. 修改 apache 配置文件

修改 apache 配置文件 /etc/httpd/conf/httpd.conf

```
User apache
Group apache
```

改成

```
User vmta
Group vmta
```

3. 修改 php session 目录权限

```
chgrp -R vmta /var/lib/php/session/
```

4. 重启 apache 服务

```
shell# /etc/init.d/httpd restart
```

4.1.2 安装配置 PostfixAdmin

当前版本为 PstfixAdmin 2.3
官方网站：http://postfixadmin.sourceforge.net/
postfixadmin 下载地址：http://sourceforge.net/projects/postfixadmin/

1. 安装 PostfixAdmin

PostfixAdmin 是用 php 写的一个 web 程序，只要将 postfixadmin 复制到 apache 的 DocumentRoot 目录下即可。

```
shell# tar zxvf postfixadmin_2.3.tar.gz -C /var/www/
shell# cd /var/www/
shell# chown -R vmta.vmta postfixadmin-2.3
shell# ln -s postfixadmin-2.3 postfixadmin
```

2. 配置mysql

```
shell# mysql -uroot -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 5
Server version: 5.0.77 Source distribution

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> CREATE DATABASE postfix;
mysql> GRANT ALL PRIVILEGES ON postfix.* TO 'pfadmin'@'localhost' IDENTIFIED BY 'pa5swd';
mysql> GRANT select ON postfix.* TO 'postfix'@'localhost' IDENTIFIED BY 'pa5swd';
mysql> flush privileges;
mysql> quit
```

3. 修改 postfixadmin 配置文件

本例的配置文件为 /var/www/postfixadmin/config.inc.php ，配置如下：

```
$CONF['configured'] = true;
$CONF['default_language'] = 'cn';

$CONF['database_type'] = 'mysql';
$CONF['database_host'] = 'localhost';
$CONF['database_user'] = 'pfadmin';
$CONF['database_password'] = 'pa5swd';
$CONF['database_name'] = 'postfix';

$CONF['admin_email'] = 'postmaster@postfixtest.com';
$CONF['encrypt'] = 'md5crypt';
$CONF['min_password_length'] = 8;
$CONF['quota'] = 'YES';
$CONF['quota_multiplier'] = '1024000';
$CONF['vacation'] = 'YES';
$CONF['user_footer_link'] = "http://change-this-to-your.domain.tld/main";
$CONF['footer_text'] = 'Return to change-this-to-your.domain.tld';
$CONF['footer_link'] = 'http://change-this-to-your.domain.tld';
$CONF['used_quotas'] = 'YES';
$CONF['new_quota_table'] = 'YES';
```

邮箱的存储格式使用domain.ltd/username的形式，所以设置：

```
$CONF['domain_path'] = 'YES';
$CONF['domain_in_mailbox'] = 'NO';
```

4.1.3 增加自动建立目录的功能

postfxiadmin不能自动创建目录，需要进行一下修改，请参考下面步骤进行修改。

1. 建立创建虚拟邮箱和删除虚拟邮箱的脚本

建立创建虚拟邮箱脚本，脚本名称 /usr/local/bin/maildir-creation.sh ，脚本内容如下：

```
#!/bin/bash
HOME_DIR="/var/vmta"
USER_NAME="vmta"
GROUP_NAME="vmta"

if [ ! -d ${HOME_DIR}/$1 ] ; then
    mkdir ${HOME_DIR}/$1
    chown -R ${USER_NAME}.${GROUP_NAME} ${HOME_DIR}/$1
fi
mkdir ${HOME_DIR}/$1/$2
chown -R ${USER_NAME}.${GROUP_NAME} ${HOME_DIR}/$1/$2
```

建立删除虚拟邮箱脚本，脚本名称 /usr/local/bin/maildir-deletion.sh ，脚本内容如下：

```
#!/bin/bash
#
# vmta ALL = NOPASSWD: /usr/local/bin/maildir-deletion.sh
#

if [ $# -ne 2 ] ; then
  exit 127
fi

DOMAIN="$1"
USER="$2"
HOME_DIR="/var/vmta"
USER_DIR="${HOME_DIR}/${DOMAIN}/${USER}"
TRASH_DIR="${HOME_DIR}/deleted-maildirs"
DATE=`date "+%Y%m%d_%H%M%S"`

if [ ! -d "${TRASH_DIR}/${DOMAIN}" ] ; then
    mkdir -p "${TRASH_DIR}/${DOMAIN}"
fi

if [ -d "${USER_DIR}" ] ; then
    mv ${USER_DIR} ${TRASH_DIR}/${DOMAIN}/${USER}-${DATE}
fi
```

建立删除目录

```
shell# mkdir /var/vmta/deleted-maildirs
shell# chown -R vmta.vmta /var/vmta/deleted-maildirs/
```

2. 赋予脚本可执行权限

```
shell# chmod 750 /usr/local/bin/maildir-creation.sh /usr/local/bin/maildir-deletion.sh
```

```
shell# chown vmta.vmta /usr/local/bin/maildir-creation.sh /usr/local/bin/maildir-deletion.sh
```

3. 配置 sudo

在 /etc/sudoers 增加一行

```
vmta ALL = NOPASSWD: /usr/local/bin/maildir-creation.sh
vmta ALL = NOPASSWD: /usr/local/bin/maildir-deletion.sh
```

在/etc/sudoers 注释掉下面内容

```
#Defaults    requiretty
```

4. 修改 postfixadmin 的相关文件

修改 create-mailbox.php 文件，225 行内容应该是：

```
db_log ($SESSID_USERNAME, $fDomain, 'create_mailbox', "$fUsername");
```

在该行前面增加下面一行：

```
system("sudo /usr/local/bin/maildir-creation.sh $fDomain ".$_POST['fUsername']);
```

修改 delete.php 文件，145 行内容应该是：

```
db_log ($SESSID_USERNAME, $fDomain, 'delete_mailbox', $fDelete);
```

在该行下面增加下面一行：

```
             $userarray=explode("@",$fDelete);
             $user=$userarray[0];
             $domain=$userarray[1];
             system("sudo /usr/local/bin/maildir-deletion.sh $domain $user");
```

4.1.4 使用 postfixadmin 安全配置

/etc/httpd/conf.d/postfixadmin.conf

```
Alias /postfixadmin "/var/www/postfixadmin/"
<Directory "/var/www/postfixadmin">
  AuthType Basic
  Authname Private
  Authuserfile /etc/httpd/conf/mailadmin.passwd
  Require valid-user
  Order deny,allow
  Allow from 192.168.1.0/24
  Deny from all
</Directory>
```

建立 apache basic 认证用户

```
shell# htpasswd -bc /etc/httpd/conf/mailadmin.passwd mailadmin pa5swd@
shell# chmod 640 /etc/httpd/conf/mailadmin.passwd
shell# chown vmta.vmta /etc/httpd/conf/mailadmin.passwd
```

重启 apache 服务

```
shell# /etc/init.d/httpd restart
```

4.1.5 使用 postfixadmin 创建虚拟域

1. 设置管理密码

访问下面地址：http://mail.yourdomain.com/postfixadmin/setup.php 设置管理密码

Create superadmin account

Setup password　　Lost password?

管理员：　　邮件地址

密码：

密码（验证）：

新增

在 setup password 后面的文本框中输入管理密码，点击“新增”按钮就会在页面上出现一行类型下面的输出：

```
$CONF['setup_password'] =
'9046ca7af7b6350b3e12f7143c6be820:0f7de2eb2a289614667e84de94711d0bf61d719c';
```

将上面这些内容复制到 /var/www/html/postfixadmin/config.inc.php 文件中对应的即可。

2. 创建 postfixadmin 管理员

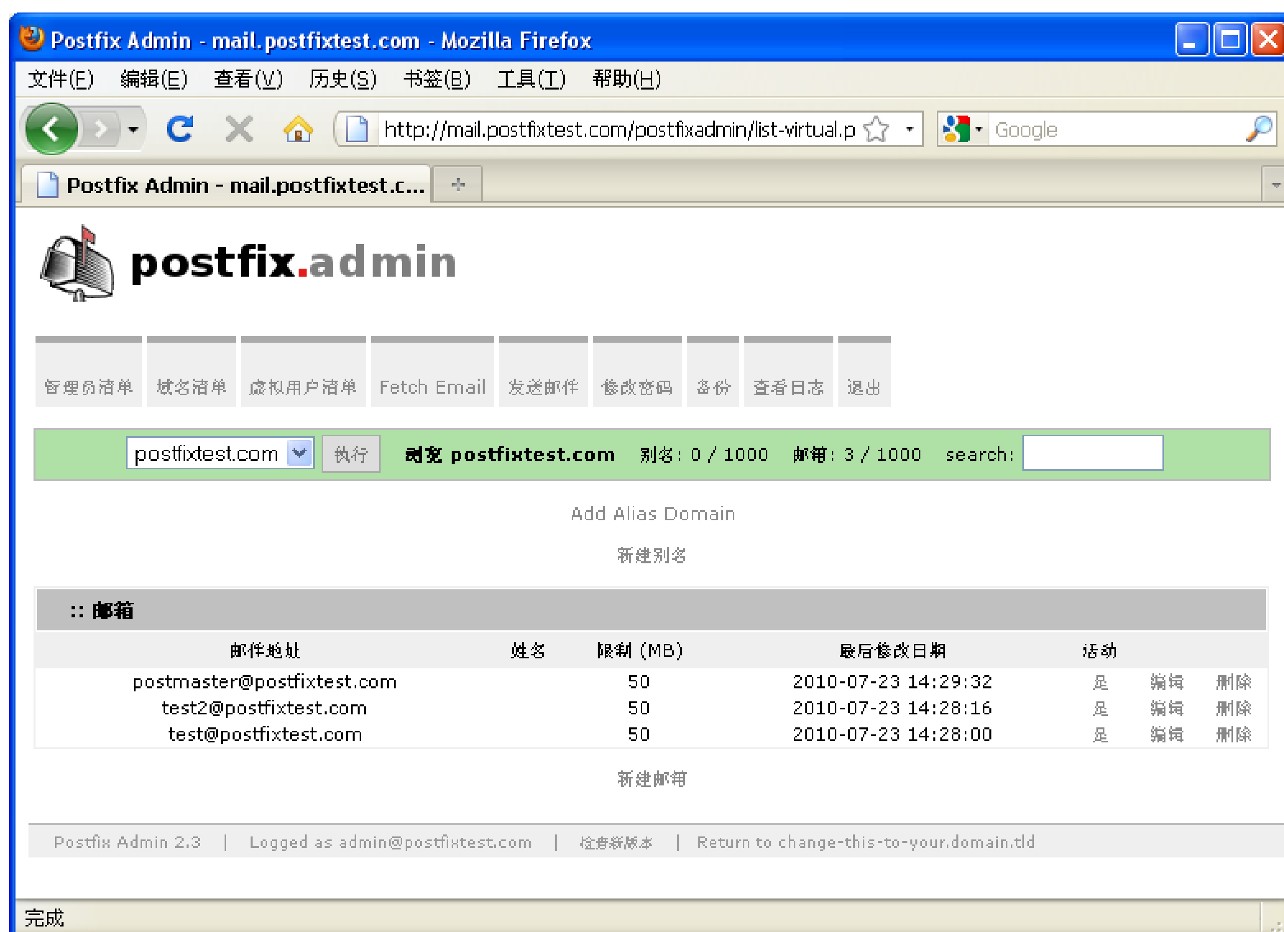
访问下面地址：http://mail.yourdomain.com/postfixadmin/setup.php ，添加管理员

3. 创建一个虚拟域并添加测试用户

postfixadmin 的管理地址： http://mail.yourdomain.com/postfixadmin/

使用 postfixadmin 创建一个虚拟域，“postfixtest.com”
增加 2 个用户 “test”，“test2”， 密码为 “123456”。
增加 postmaster 用户，密码一定复杂一些。

4. postfixadmin 的界面



4.2 配置 dovecot 支持虚拟域

4.2.1 配置 /etc/dovecot.conf

dovecot 的配置文件为/etc/dovecot.conf

备份原有文件：

```
shell# cd /etc/
shell# cp dovecot.conf dovecot.conf.dist
```

修改/etc/dovecot.conf

```
protocols = pop3 imap
first_valid_uid = 5000
mail_uid = 5000
mail_gid = 5000

### Logging verbosity
#auth_verbose=yes
#auth_debug=yes
#auth_debug_passwords=yes
#mail_debug=yes

disable_plaintext_auth = no

log_timestamp = "%Y-%m-%d %H:%M:%S "
mail_location = maildir:/var/vmta/%d/%n
#mail_location = maildir:~/Maildir
#log_path = /var/log/dovecot.log

### add defualt domain
auth_default_realm = postfixtest.com

protocol lda {
  mail_plugins = quota
  log_path = /var/vmta/dovecot-deliver.log
  info_log_path = /var/vmta/dovecot-deliver.log
  auth_socket_path = /var/run/dovecot/auth-master
  postmaster_address = postmaster@postfixtest.com
}
auth default {
  mechanisms = plain login
  userdb sql {
     args = /etc/dovecot-mysql.conf
  }
  userdb prefetch {
  }
  passdb sql {
     args = /etc/dovecot-mysql.conf
  }
  socket listen {
    master {
      path = /var/run/dovecot/auth-master
      mode = 0600
      user = vmta
      group = vmta
    }
    client {
      path = /var/spool/postfix/private/auth
```

```
      mode = 0660
      user = postfix
      group = postfix
    }
  }
}
```

4.2.2 配置 /etc/dovecot-mysql.conf

默认/etc/dovecot-mysql.conf 是不存在的，建立 /etc/dovecot-mysql.conf

```
driver=mysql
connect = host=localhost dbname=postfix user=postfix password=pa5swd
default_pass_scheme = MD5
### config for virtual
user_query = SELECT concat('/var/vmta/',maildir) as home , 5000 AS uid , 5000 AS gid FROM mailbox WHERE username = '%u'

password_query = SELECT password , concat('/var/vmta/',maildir) as home , 5000 AS uid , 5000 AS gid FROM mailbox WHERE username = '%u'
```

修改 /etc/dovecot.conf 和 /etc/dovecot-mysql.conf 权限

```
shell# chmod 644 /etc/dovecot.conf /etc/dovecot-mysql.conf
```

重启 dovecot 服务

```
shell# /etc/init.d/dovecot restart
```

可能会出现下面信息：

```
Restarting mail server: dovecotWarning: Fixing permissions of /var/run/dovecot to be world-readable
Warning: Corrected permissions for login directory /var/run/dovecot/login
```

确认 /var/run/dovecot 目录有没有生成，如果生成了下面的目录，说明配置正确了：

```
shell# cd /var/run/dovecot
shell# ls -l
total 8
srw------- 1 vmta vmta       0 Jul 23 14:46 auth-master
srw------- 1 root root       0 Jul 23 14:46 auth-worker.4702
srwxrwxrwx 1 root root       0 Jul 23 14:46 dict-server
lrwxrwxrwx 1 root root      17 Jul 23 14:46 dovecot.conf -> /etc/dovecot.conf
drwxr-x--- 2 root dovecot 4096 Jul 23 14:46 login
-rw------- 1 root root       5 Jul 23 14:46 master.pid
```

检查/var/log/mail.log 是否存在错误，如果有错误，请检查配置。

测试 Dovecot 配置文件是否正确：

```
shell# /usr/libexec/dovecot/authtest test@postfixtest.com
```

输出结果：

```
shell# /usr/libexec/dovecot/authtest test@postfixtest.com
userdb: test@postfixtest.com
uid   : 5000
gid   : 5000
user  : test@postfixtest.com
home  : /var/vmta/postfixtest.com/test/
```

4.3 配置 postfix 支持虚拟域

4.3.1 让 postfix 使用 dovecot LDA 投递虚拟域邮件

修改/etc/postfix/master.cf

```
dovecot   unix  -       n       n       -       -       pipe
    flags=DRhu user=vmta:vmta argv=/usr/libexec/dovecot/deliver -d ${recipient}
```

4.3.2 让 postfix 支持 stmp 认证

修该/etc/postfix/main.cf 文件,增加如下内容：

```
### banner setting
mail_name = sendmail 8.11.6+Sun/8.11.6
smtpd_banner = ESMTP $mail_name
### defind external command path
debugger_command = PATH=/bin:/usr/bin:/usr/local/bin
### mynetworks setting
mynetworks = 127.0.0.0/8
### disable vrfy
disable_vrfy_command = yes
### virtural user setting
virtual_alias_maps = mysql:/etc/postfix/mysql_virtual_alias_maps.cf
virtual_mailbox_domains = mysql:/etc/postfix/mysql_virtual_domains_maps.cf
virtual_mailbox_maps = mysql:/etc/postfix/mysql_virtual_mailbox_maps.cf
smtpd_sender_login_maps = mysql:/etc/postfix/mysql_virtual_sender_maps.cf
virtual_mailbox_base = /var/vmta
virtual_gid_maps = static:5000
virtual_uid_maps = static:5000
virtual_transport = dovecot
dovecot_destination_recipient_limit = 1
### smtp auth setting
smtpd_sender_restrictions = permit_mynetworks,
        reject_authenticated_sender_login_mismatch
smtpd_recipient_restrictions = permit_mynetworks,
        permit_sasl_authenticated,
        reject_non_fqdn_hostname,
```

```
        reject_non_fqdn_sender,
        reject_non_fqdn_recipient,
        reject_unauth_destination,
        reject_unauth_pipelining,
        reject_invalid_hostname
broken_sasl_auth_clients = yes
smtpd_sasl_type = dovecot
smtpd_sasl_path = private/auth
smtpd_sasl_auth_enable = yes
smtpd_sasl_local_domain = $myhostname
smtpd_sasl_security_options = noanonymous
```

4.3.3 配置虚拟域

建立虚拟域配置文件
文件名：/etc/postfix/mysql_virtual_domains_maps.cf

```
user = postfix
password = pa5swd
hosts  = 127.0.0.1
dbname = postfix
query  = SELECT domain FROM domain WHERE domain='%s' and active = '1'
```

建立虚拟邮箱配置文件
文件名：/etc/postfix/mysql_virtual_mailbox_maps.cf

```
user = postfix
password = pa5swd
hosts  = 127.0.0.1
dbname = postfix
query  = SELECT maildir FROM mailbox WHERE username='%s' AND active = '1'
```

注建立虚拟邮箱别名配置文件
文件名：/etc/postfix/mysql_virtual_alias_maps.cf

```
user = postfix
password = pa5swd
hosts = 127.0.0.1
dbname = postfix
query  = SELECT goto FROM alias WHERE address='%s' AND active = '1'
```

建立发件人邮箱检验配置文件，确保发件人不能伪造其他人的邮件地址发送邮件
文件名：/etc/postfix/mysql_virtual_sender_maps.cf

```
user = postfix
password = pa5swd
hosts = 127.0.0.1
dbname = postfix
query = SELECT username FROM mailbox WHERE username='%s' AND active='1'
```

注意：以上几个文件中如果连接本地的 mysql 数据库，hosts 的值必须为 127.0.0.1，不能使用

localhost，否则会出错。

测试配置文件是否正确：

```
shell# postmap -q "test@postfixtest.com" mysql:/etc/postfix/mysql_virtual_mailbox_maps.cf
postfixtest.com/test/
```

```
shell# postmap -q "postfixtest.com" mysql:/etc/postfix/mysql_virtual_domains_maps.cf
postfixtest.com
```

```
shell# postmap -q "test@postfixtest.com" mysql:/etc/postfix/mysql_virtual_sender_maps.cf
test@postfixtest.com
```

重启动 postfix 服务

```
shell# /etc/init.d/postfix restart
```

检查 /var/log/maillog 是否存在错误，如果有错误，请检查配置。

4.4 使用虚拟用户进行收发邮件测试

4.4.1 建立转换 base64 编码程序

建立 /usr/local/bin/encode_base64.pl 脚本

```
#!/usr/bin/perl -w
use strict;
use MIME::Base64;
print "Input you string.\n";
print "Press Enter to exit.\n";
print "=> ";
my $line;
while( $line = <STDIN> )
{
    if( $line ne "\n" ){
       chomp($line) ;
       print "Base64 encode is : " . encode_base64($line) ;
       print "=> ";
    }
    else
    {
      exit;
    }
}
```

```
shell# chmod 750 /usr/local/bin/encode_base64.pl
```

生成 test 用户的 base64 编码

```
shell# encode_base64.pl
Input you string.
Press Enter to exit.
=> test@postfixtest.com
Base64 encode is : dGVzdEBwb3N0Zml4dGVzdC5jb20=
=> 123456
Base64 encode is : MTIzNDU2
=>
```

4.4.2 验证 smtp 认证是否配置正确

```
shell# telnet localhost 25
Escape character is '^]'.
220 ESMTP sendmail 8.11.6+Sun/8.11.6
ehlo localhost
250-red.postfix.internal
250-PIPELINING
250-SIZE 10240000
250-ETRN
250-AUTH PLAIN LOGIN
250-AUTH=PLAIN LOGIN
250-ENHANCEDSTATUSCODES
250-8BITMIME
250 DSN
auth login
334 VXNlcm5hbWU6
dGVzdEBwb3N0Zml4dGVzdC5jb20=
334 UGFzc3dvcmQ6
MTIzNDU2
235 2.7.0 Authentication successful
quit
221 2.0.0 Bye
Connection closed by foreign host.
```

4.4.3 测试 pop3 服务配置是否正确

```
shell# telnet localhost 110
Trying 127.0.0.1...
Connected to localhost.localdomain (127.0.0.1).
Escape character is '^]'.
+OK Dovecot ready.
user test@postfixtest.com
+OK
pass 123456
+OK Logged in.
list
+OK 1 messages:
1 603
.
retr 1
+OK 603 octets
```

```
Return-Path: <admin@postfixtest.com>
Delivered-To: test@postfixtest.com
Received: from 192.168.1.70 (localhost.localdomain [127.0.0.1])
        by red.postfix.internal (Postfix) with ESMTP id 617339C755
        for <test@postfixtest.com>; Fri, 23 Jul 2010 14:28:01 +0800 (CST)
To: test@postfixtest.com
From: admin@postfixtest.com
Subject: =?utf-8?Q?=E6=AC=A2=E8=BF=8E?=
MIME-Version: 1.0
Content-Type: text/plain; charset=utf-8
Content-Transfer-Encoding: 8bit
Message-Id: <20100723062801.617339C755@red.postfix.internal>
Date: Fri, 23 Jul 2010 14:28:01 +0800 (CST)

Hi,

Welcome to your new account.
.
quit
+OK Logging out.
Connection closed by foreign host.
```

测试完毕后说明已经支持虚拟域了。

4.5 使用虚拟用户邮箱配额限制

4.5.1 配置 Dovecot

1. 修改 /etc/dovecot.conf ，添加如下内容：

```
### virtaul user quota
protocol imap {
  mail_plugins = quota imap_quota
}
protocol pop3 {
  mail_plugins = quota
}
plugin {
  ### v1.2 + SQL:
  quota = maildir
  quota_rule = *:storage=100M:messages=1000
  ###
  quota_warning3 = storage=90%% /usr/local/bin/quota-warning.sh 90 %u
  quota_warning2 = storage=80%% /usr/local/bin/quota-warning.sh 80 %u
  quota_warning  = storage=70%% /usr/local/bin/quota-warning.sh 70 %u
}
```

2. 修改 /etc/dovecot-mysql.conf，添加如下内容：

```
driver=mysql
connect = host=localhost dbname=postfix user=postfix password=pa5swd
```

```
default_pass_scheme = MD5
### config for virtual user with quotas
user_query = SELECT concat('/var/vmta/',maildir) AS home , 5000 AS uid , 5000 AS gid,
CONCAT('*:bytes=', CAST(quota AS CHAR)) AS quota_rule FROM mailbox WHERE username = '%u'
AND active = '1'

password_query = SELECT password , concat('/var/vmta/',maildir) as home , 5000 AS uid ,
5000 AS gid FROM mailbox WHERE username = '%u'
```

3. 重启动 Doveoct

```
shell# /etc/init.d/dovecot restart
```

4.5.2 建立 quota-warning 脚本

建立 /usr/local/bin/quota-warning.sh 文件

```
#!/bin/bash

PERCENT=$1
USER=$2
FROM="postmaster@postfixtest.com"
QWF="/tmp/quota.warning.$$"

echo "From: $FROM
To: $USER,$FROM
Subject: Your email quota is $PERCENT% full
Content-Type: text/plain; charset=\"UTF-8\"

This is an automatic message to warn that your mailbox is now $PERCENT% full." > $QWF

/usr/sbin/sendmail -t  < $QWF
rm -f $QWF

exit 0
```

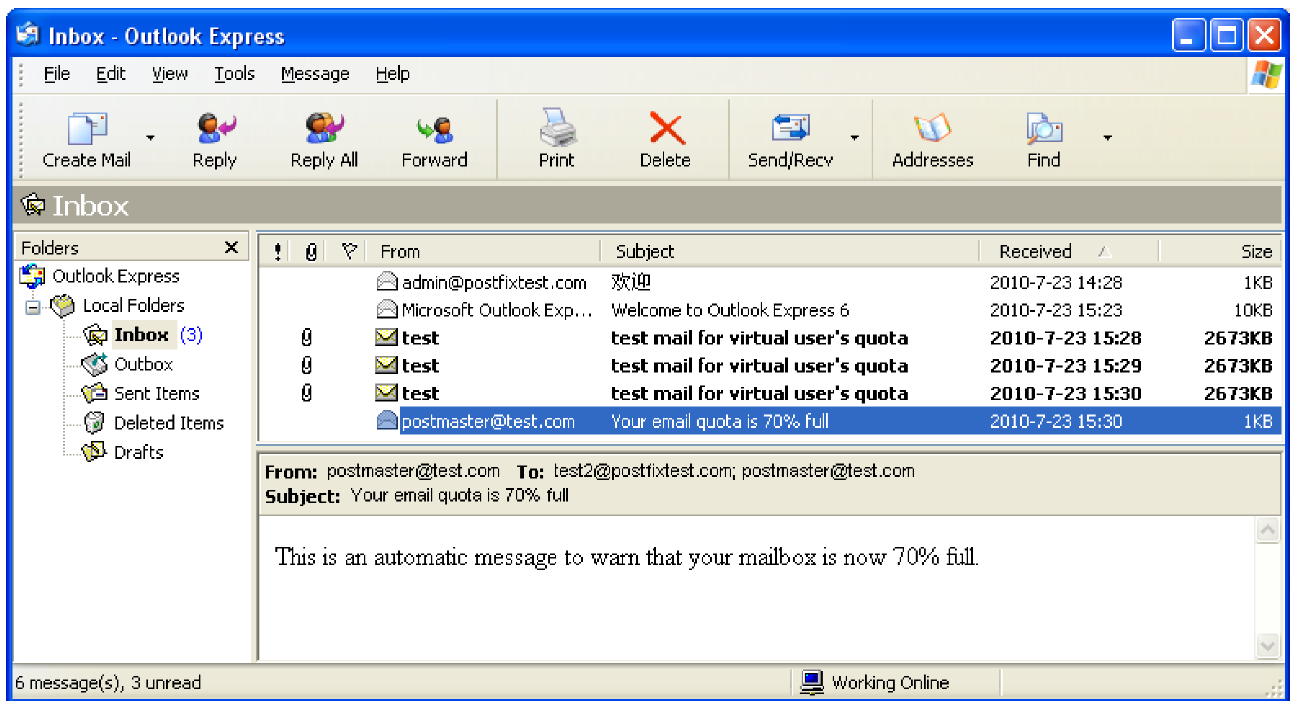
修改 /usr/local/bin/quota-warning.sh 脚本权限

```
shell# chmod 750 /usr/local/bin/quota-warning.sh
```

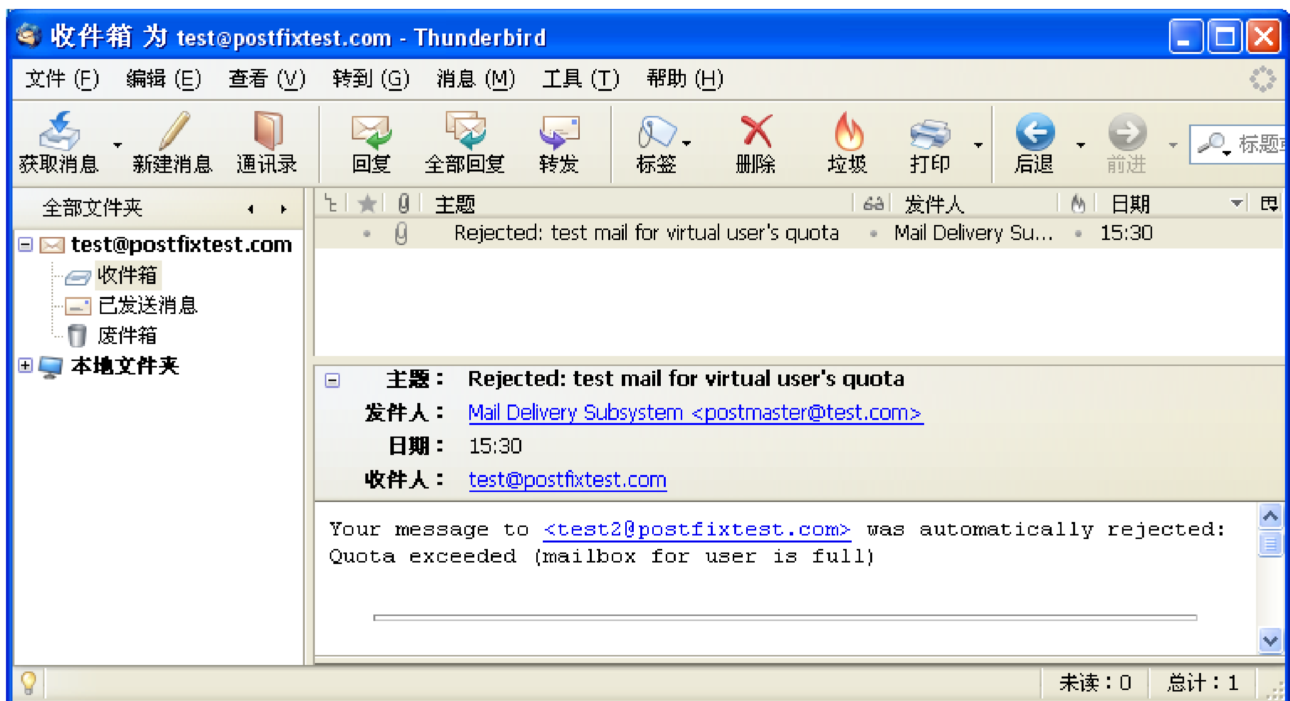
4.5.3 虚拟用户配额测试

用 test 用户给 test2 用户发送邮件，测试配额是否生效。通过 postfixamdin 修改 test2 用户的最大邮箱为 10M，然后进行测试。

Test2 用户收到了邮箱超过 70% 的提醒。



test 用户收到了对方邮箱满了，投递失败的退信信息。



说明：如果没有合适的文件可以使用 dd 生成 1M 的文件：

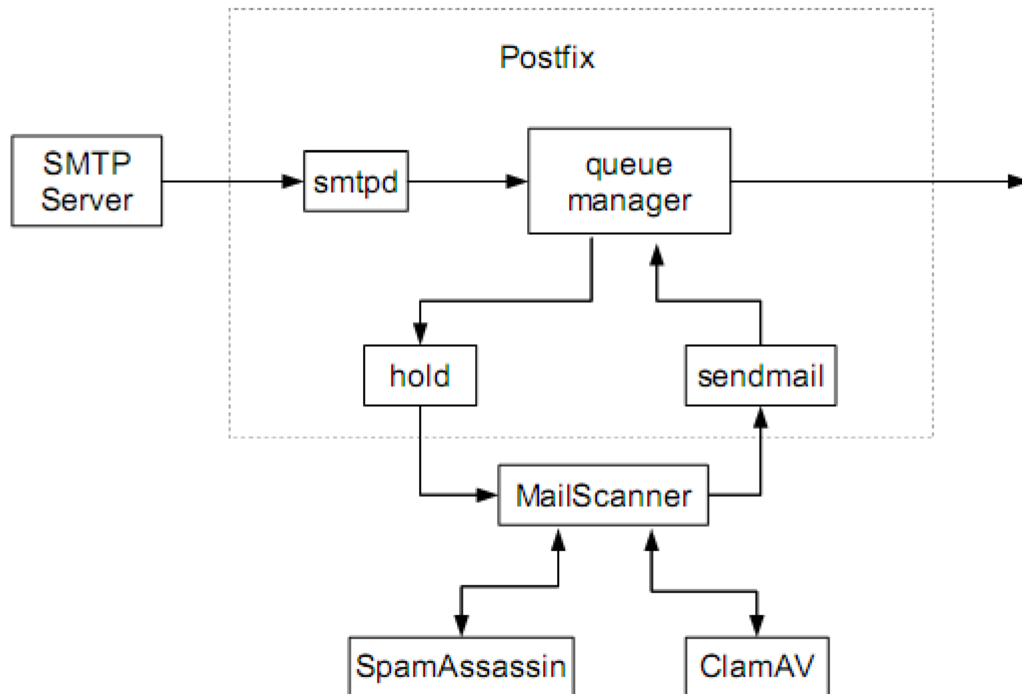
```
shell# dd if=/dev/zero of=size1m.txt bs=1024000 count=1
```

4.6 小结

经过以上的配置，一套虚拟用户邮件系统已经搭建完成。

五. 安装反病毒和反垃圾邮件模块

下面是邮件系统与防病毒、反垃圾邮件之间的关系图，如下图：



Postfix 与 垃圾邮件防护模块之间的关系：

1. postfix 收到邮件后将邮件放到 hold 队列中。
2. MailScanner 定期检查 postfix 的 hold 队列中的邮件。
3. MailScanner 将邮件交给 ClamAV 和 SpamAssassin 检查是否为病毒邮件或者垃圾邮件。
4. Mailscanner 收到 ClamAV 和 SpamAssassin 反馈结果查看是否为正常邮件，如果是正常邮件交给 postfix 进行处理，如果不是正常邮件则按照 MailScanner 的配置进行处理。

5.1 安装 Clamav

Clam AntiVirus 是一款 UNIX 下开源的 (GPL) 反病毒工具包，专为邮件网关上的电子邮件扫描而设计。该工具包提供了包含灵活且可伸缩的监控程序、命令行扫描程序以及用于自动更新数据库的高级工具在内的大量实用程序。该工具包的核心在于可用于各类场合的反病毒引擎共享库。

官方网站：http://www.clamav.net/

5.1.1 下载 ClamAV rpm 包

下载地址：http://packages.sw.be/clamav/
http://atrpms.net/dist/el5/clamav/
http://www.clamav.net/download/packages/packages-linux

5.1.2 安装Clamav

```
shell# rpm -ivh clamav-db-0.96.1-1.el5.rf.i386.rpm
shell# rpm -ivh clamav-0.96.1-1.el5.rf.i386.rpm
shell# rpm -ivh clamd-0.96.1-1.el5.rf.i386.rpm
```

5.2 安装 SpamAssassin模块

SpamAssassin是一套邮件内容过滤软件，是用perl编写的。现在由Apache基金会维护。想了解更多内容，请参阅官方网站。

官方网站：http://spamassassin.apache.org/

5.2.1 下载 SpamAssassin 源码

下载地址：http://spamassassin.apache.org/downloads.cgi

从官方网站下载相应版本软件包。

5.2.2 安装perl模块

```
shell# yum install -y perl-Net-DNS perl-Digest-HMAC perl-HTML-Parser perl-HTML-Tagset
shell# yum install -y perl-Archive-Tar perl-IO-Zlib perl-libwww-perl perl-Digest-SHA1
```

5.2.3 生成 SpamAssassin 的rpm包

```
shell# rpmbuild -ta Mail-SpamAssassin-3.2.5.tar.gz
```

5.2.4 安装 SpamAssassin

```
shell# cd /usr/src/redhat/RPMS/i386
shell# rpm -ivh perl-Mail-SpamAssassin-3.2.5-1.i386.rpm
shell# rpm -vih spamassassin-3.2.5-1.i386.rpm
```

注意：如果出现下面错误

```
Processing files: perl-Mail-SpamAssassin-3.2.5-1
error: File not found: /var/tmp/spamassassin-root/usr/share/spamassassin
Checking for unpackaged file(s): /usr/lib/rpm/check-files /var/tmp/spamassassin-root
warning: Installed (but unpackaged) file(s) found:
   /usr/lib/perl5/5.8.8/i386-linux-thread-multi/perllocal.pod


RPM build errors:
    Bad exit status from /var/tmp/rpm-tmp.64468 (%doc)
    File not found: /var/tmp/spamassassin-root/usr/share/spamassassin
    Installed (but unpackaged) file(s) found:
   /usr/lib/perl5/5.8.8/i386-linux-thread-multi/perllocal.pod
```

修改spamassasin.spec文件的107行后，增加

修改前：

```
mkdir -p %{buildroot}/etc/mail/spamassassin

[ -x /usr/lib/rpm/brp-compress ] && /usr/lib/rpm/brp-compress
```

修改后：

```
mkdir -p %{buildroot}/etc/mail/spamassassin

mkdir -p %{buildroot}/usr/share/spamassassin
install -t %buildroot/%{_datadir}/spamassassin rules/*
[ -x /usr/lib/rpm/brp-compress ] && /usr/lib/rpm/brp-compress
```

5.3 安装 MailScanner 模块

5.3.1 下载MailScanner

官方网站：www.mailscanner.info
去官方网站下载适MailScanner的稳定版。

5.3.2 安装 MailScanner

```
shell# tar zxvf MailScanner-4.79.11-1.rpm.tar.gz
shell# cd MailScanner-4.79.11-1
shell# ./install.sh
```

5.4 配置反垃圾邮件和反病毒模块

5.4.1 修改postfix配置文件

修改 /etc/postfix/main.cf

```
### add for MailScanner
header_checks = regexp:/etc/postfix/header_checks
```

调试阶段将soft_bounce 打开，以免造成“误伤”。

```
soft_bounce = yes
```

修改 /etc/postfix/header_checks

```
# add for MailScanner
/^Received:/ HOLD
```

5.4.2 修改MailScanner配置文件

/etc/MailScanner/MailScanner.conf

1. 基本配置

```
Max Children = 5
Run As User = postfix
Run As Group = postfix
Queue Scan Interval = 3
Incoming Queue Dir = /var/spool/postfix/hold
Outgoing Queue Dir = /var/spool/postfix/incoming
MTA = postfix
Sendmail = /usr/sbin/sendmail
Sign Clean Messages = no
```

```
%org-name% = yoursite                   #请修改
%web-site% = www.your-organisation.com   #请修改
```

2. SA 相关配置

```
Required SpamAssassin Score = 3
High SpamAssassin Score = 20
#满足 Required SpamAssassin Score 的动作，投递并在邮件头中打上标记
Spam Actions = deliver header "X-Spam-Status: Yes"
#高分动作，存储
High Scoring Spam Actions = store
SpamAssassin User State Dir = /var/spool/MailScanner/spamassassin
#标记为垃圾邮件的标题标记前缀
Spam Subject Text = ****Spam***
#标记为附件有问题的邮件标题标记前缀
Filename Subject Text = {Filename?}
#在信头中的分数级别显示的符号
Spam Score Character = +
```

```
Mail Header = X-%org-name%-MailScanner:

# Add this extra header to all messages found to be spam.
# This can also be the filename of a ruleset.
Spam Header = X-%org-name%-MailScanner-SpamCheck:

# Add this extra header if "Spam Score" = yes. The header will
# contain 1 character for every point of the SpamAssassin score.
Spam Score Header = X-%org-name%-MailScanner-SpamScore:
```

3. Virus Scan 相关配置

```
Virus Scanners = clamav
```

4. 调试阶段可以将日志打开

```
Log Speed = yes
```

```
Log Spam = yes
```

5. 修改信头标记
/etc/MailScanner/spam.assassin.prefs.conf

```
envelope_sender_header X-yourname-MailScanner-From
```

6. 配置自动升级 ClamAV

MailScanner 会自动升级 clamav 的病毒库，程序有点小问题，在 /var/log/maillog 中会报如下错误：

```
Jul 23 15:10:44 red ClamAV-autoupdate[2971]: ClamAV updater /usr/local/bin/freshclam
cannot be run
```

修改 /usr/lib/MailScanner/clamav-autoupdate 文件，将下面一行：

```
$ClamUpdateCommand = "$PackageDir/bin/freshclam";
```

改成：

```
$ClamUpdateCommand = "/usr/bin/freshclam";
```

5.4.3 修改/var/spool/MailScanner 目录权限

```
shell# chown -R postfix:postfix /var/spool/MailScanner/*
```

5.4.4 启动相关服务

启动 SpamAssasin

```
shell# /etc/init.d/spamassassin start
```

启动 Clamav

```
shell# /etc/init.d/clamd start
```

启动 MailScanner

```
shell# /etc/init.d/MailScanner start
```

启动器查看/var/log/maillog 是否有错误。

如果有错误可以使用下面命令来检查错误相关信息：

```
shell# MailScanner --lint
```

5.4.5 验证 MailScanner 是否正常

1. 检验反垃圾邮件是否正常

spam 安装后自带一个垃圾邮件的例子，在 /usr/share/doc/spamassassin-3.2.5 目录下。

```
shell# cd /usr/share/doc/spamassassin-3.2.5
shell# sendmail test@postfixtest.com < sample-spam.txt
```

/var/log/maillog 里面日志

```
Jul 23 16:04:01 red postfix/pickup[5747]: A3C409C75E: uid=0 from=<root>
Jul 23 16:04:01 red postfix/cleanup[5778]: A3C409C75E: hold: header Received: by
red.postfix.internal (sendmail 8.11.6+Sun/8.11.6, from userid 0)??id A3C409C75E; Fri, 23
Jul 2010 16:04:01 +0800 (CST) from local; from=<root@red.postfix.internal>
to=<test@postfixtest.com>
Jul 23 16:04:01 red postfix/cleanup[5778]: A3C409C75E: message-
id=<GTUBE1.1010101@example.net>
Jul 23 16:04:02 red MailScanner[5772]: New Batch: Scanning 1 messages, 1189 bytes
Jul 23 16:04:03 red MailScanner[5772]: Virus and Content Scanning: Starting
Jul 23 16:04:16 red MailScanner[5772]: Virus Scanning completed at 87 bytes per second
Jul 23 16:04:16 red MailScanner[5772]: Spam Checks: Starting
Jul 23 16:04:16 red MailScanner[5772]: SpamAssassin cache hit for message A3C409C75E.A4D6C
Jul 23 16:04:16 red MailScanner[5772]: Message A3C409C75E.A4D6C from 127.0.0.1
(root@red.postfix.internal) to postfixtest.com is spam, SpamAssassin (cached,
score=1004.75, required 3, DATE_IN_PAST_96_XX 2.32, DNS_FROM_OPENWHOIS 2.43, GTUBE
1000.00, NO_RELAYS -0.00)
Jul 23 16:04:16 red MailScanner[5772]: Spam Checks: Found 1 spam messages
Jul 23 16:04:16 red MailScanner[5772]: Spam Actions: message A3C409C75E.A4D6C actions are
store
Jul 23 16:04:16 red MailScanner[5772]: Spam Checks completed at 41180 bytes per second
Jul 23 16:04:16 red MailScanner[5772]: Deleted 1 messages from processing-database
Jul 23 16:04:16 red MailScanner[5772]: Batch completed at 87 bytes per second (1189 / 13)
Jul 23 16:04:16 red MailScanner[5772]: Batch (1 message) processed in 13.62 seconds
```

这个垃圾邮件别放到隔离区了，可以到 /var/spool/MailScanner/quarantine 目录下查找，在调试的是候是有用的。

```
shell# ls -l /var/spool/MailScanner/quarantine/20100723/spam
total 4
-rw------- 1 postfix postfix 931 Jul 23 16:04 A3C409C75E.A4D6C
```

测试非垃圾邮件例子

```
shell# sendmail test@postfixtest.com < sample-nonspam.txt
```

/var/log/maillog 里面日志

```
Jul 23 16:06:49 red postfix/pickup[5747]: 97C749C75E: uid=0 from=<root>
Jul 23 16:06:49 red postfix/cleanup[5814]: 97C749C75E: hold: header Received: by
red.postfix.internal (sendmail 8.11.6+Sun/8.11.6, from userid 0)??id 97C749C75E; Fri, 23
Jul 2010 16:06:49 +0800 (CST) from local; from=<root@red.postfix.internal>
to=<test@postfixtest.com>
```

```
Jul 23 16:06:49 red postfix/cleanup[5814]: 97C749C75E: message-id=<v0421010eb70653b14e06@[208.192.102.193]>
Jul 23 16:06:49 red MailScanner[5772]: New Batch: Scanning 1 messages, 6961 bytes
Jul 23 16:06:49 red MailScanner[5772]: Virus and Content Scanning: Starting
Jul 23 16:07:03 red MailScanner[5772]: Virus Scanning completed at 516 bytes per second
Jul 23 16:07:03 red MailScanner[5772]: Spam Checks: Starting
Jul 23 16:07:07 red MailScanner[5772]: Message 97C749C75E.A7A17 from 127.0.0.1 (root@red.postfix.internal) to postfixtest.com is spam, SpamAssassin (not cached, score=12.298, required 6, DNS_FROM_OPENWHOIS 2.43, URIBL_BLACK 1.96, URIBL_GREY 0.25, URIBL_OB_SURBL 2.13, URIBL_RHS_DOB 0.90, URIBL_SC_SURBL 2.52, URIBL_WS_SURBL 2.10)
Jul 23 16:07:07 red MailScanner[5772]: Spam Checks: Found 1 spam messages
Jul 23 16:07:07 red MailScanner[5772]: Spam Actions: message 97C749C75E.A7A17 actions are deliver,header
Jul 23 16:07:07 red MailScanner[5772]: Spam Checks completed at 1790 bytes per second
Jul 23 16:07:07 red MailScanner[5772]: Requeue: 97C749C75E.A7A17 to 97E209C75C
Jul 23 16:07:07 red postfix/qmgr[5746]: 97E209C75C: from=<root@red.postfix.internal>, size=6731, nrcpt=1 (queue active)
Jul 23 16:07:07 red MailScanner[5772]: Uninfected: Delivered 1 messages
Jul 23 16:07:07 red MailScanner[5772]: Virus Processing completed at 267000 bytes per second
Jul 23 16:07:07 red MailScanner[5772]: Deleted 1 messages from processing-database
Jul 23 16:07:07 red MailScanner[5772]: Batch completed at 399 bytes per second (6961 / 17)
Jul 23 16:07:07 red MailScanner[5772]: Batch (1 message) processed in 17.43 seconds
Jul 23 16:07:07 red postfix/pipe[5827]: 97E209C75C: to=<test@postfixtest.com>, relay=dovecot, delay=18, delays=18/0.14/0/0.27, dsn=2.0.0, status=sent (delivered via dovecot service)
Jul 23 16:07:07 red postfix/qmgr[5746]: 97E209C75C: removed
```

信头中的信息

```
X-PostfixTest-MailScanner-Information: Please contact your mail administartor for more information
X-PostfixTest-MailScanner-ID: 97C749C75E.A7A17
X-PostfixTest-MailScanner: Found to be clean
X-PostfixTest-MailScanner-SpamCheck: spam, SpamAssassin (not cached,
    score=12.298, required 6, DNS_FROM_OPENWHOIS 2.43, URIBL_BLACK 1.96,
    URIBL_GREY 0.25, URIBL_OB_SURBL 2.13, URIBL_RHS_DOB 0.90,
    URIBL_SC_SURBL 2.52, URIBL_WS_SURBL 2.10)
X-PostfixTest-MailScanner-SpamScore: ++++++++++++
X-PostfixTest-MailScanner-From: root@red.postfix.internal
X-Spam-Status: Yes
```

2. 检验反病毒功能是否正常

使用客户端，发送一个带附件的邮件，附件的扩展名用 “.bat” ， “.exe” ， “.com” 等 Windows 执行文件的扩展名。

```
Jul 23 16:25:20 red MailScanner[6159]: New Batch: Scanning 1 messages, 3399 bytes
Jul 23 16:25:20 red MailScanner[6159]: Filename Checks: Possible malicious batch file script (A492D9C755.A433F Spam  test.bat)
Jul 23 16:25:20 red MailScanner[6159]: Other Checks: Found 1 problems
```

```
Jul 23 16:25:20 red MailScanner[6159]: Virus and Content Scanning: Starting
Jul 23 16:25:34 red MailScanner[6159]: Virus Scanning completed at 233 bytes per second
Jul 23 16:25:34 red MailScanner[6159]: Saved infected "Spam  test.bat" to
/var/spool/MailScanner/quarantine/20100723/A492D9C755.A433F
```

系统将这个附件放到隔离区，同时给发件人和收件人都发送一封邮件，提示这封邮件有问题。

5.5 安装 MailWatch

经过上面的配置，已经满足邮件过滤的功能，不过有没有一个可视化的报告页面呢？答案是肯定的，那就是 MailWatch 。MailWatch 是为 MailScreen 编写的一个 WEB 前端界面，有了 MailWatch 我们可以查看 MailScanner 的动作，比如，垃圾邮件，病毒邮件。可以跟商业的垃圾邮件网关相媲美，而且我们还能看到源码，这是商业垃圾邮件网关无法比拟的。

MailWatch 是 sourceforge 的一个项目，下载地址：http://mailwatch.sourceforge.net/，下载相应的版本。

5.5.1 基础软件配置

修改 php 配置文件：/etc/php.ini

```
register_globals = Off
magic_quotes_gpc = On
magic_quotes_runtime = Off
session.auto_start = 0
```

MailWatch 需要 php-gd 模块支持，因此，没有安装 php-gd 请用下面命令安装 php-gd 模块。

```
shell# yum install -y php-gd
```

5.5.2 安装 MailWatch

1. 解压软件

```
shell# tar zxvf mailwatch-1.0.5.tar.gz -C /usr/local/
shell# cd /usr/local/
shell# ln -s mailwatch-1.0.5/ mailwatch
```

2. 数据库相关设置

初始化数据库

```
shell# mysql -uroot -p < /usr/local/mailwatch/create.sql
```

建立 mysql 用户

```
shell# mysql -u root -p
mysql> GRANT ALL ON mailscanner.* TO mailwatch@localhost IDENTIFIED BY 'pa5swd';
mysql> GRANT FILE ON *.* TO mailwatch@localhost IDENTIFIED BY 'pa5swd';
```

```
mysql> FLUSH PRIVILEGES;
```

建立 web 管理用户

```
shell# mysql mailscanner -u mailwatch -p
Enter password: ******
mysql> use mailscanner;
mysql> INSERT INTO users (username,password,type) VALUES ('admin',md5('pa5swd'),'A');
```

3. MailWatch 相关配置

将 MailWatch.pm 复制到 /usr/lib/MailScanner/MailScanner/CustomFunctions 目录中，对于非 rpm 安装的复制到 /opt/MailScanner/lib/MailScanner/MailScanner/CustomFunctions 目录中。

```
shell# cd /usr/local/mailwatch
shell# cp MailWatch.pm /usr/lib/MailScanner/MailScanner/CustomFunctions/
```

修改 /usr/lib/MailScanner/MailScanner/CustomFunctions/MailWatch.pm 文件，将 $db_user 和 $db_pass 改成建立好 mysql 用户和密码：

将下面内容：

```
my($db_user) = 'root';
my($db_pass) = '';
```

改成：

```
my($db_user) = 'mailwatch';
my($db_pass) = 'pa5swd';
```

4. 黑名单/白名单设置

```
shell# cp SQLBlackWhiteList.pm /usr/lib/MailScanner/MailScanner/CustomFunctions/
```

修改 /usr/lib/MailScanner/MailScanner/CustomFunctions/SQLBlackWhiteList.pm 文件

```
my($db_user) = 'root';
my($db_pass) = '';
```

改成：

```
my($db_user) = 'mailwatch';
my($db_pass) = 'pa5swd';
```

修改 /etc/MailScanner/spam.assassin.prefs.conf

```
bayes_path /etc/MailScanner/bayes/bayes
bayes_file_mode 0660
bayes_auto_learn 0
```

建立目录，赋权限

```
shell# mkdir /etc/MailScanner/bayes
shell# chown root:vmta /etc/MailScanner/bayes
shell# chmod g+rws /etc/MailScanner/bayes
```

测试

```
spamassassin -D -p /etc/MailScanner/spam.assassin.prefs.conf --lint
```

5. Web 相关配置

```
shell# cd /usr/local/mailwatch
shell# mv mailscanner/ /var/www/html/
shell# chown -R vmta.vmta /var/www/html/mailscanner/
shell# chmod -R ug+rwx /var/www/html/mailscanner/images
```

配置 apache ，建立 /etc/httpd/conf.d/mailwatch.conf

```
#Alias /mailscanner "/usr/local/mailwatch/mailscanner"
<Directory "/var/www/html/mailscanner/">
  Order deny,allow
  Allow from 192.168.1.0/24 10.4.0.0/16
  Deny from all
</Directory>
```

修改 mailwatch web 配置文件

将 /var/www/html/mailscanner/conf.php.example 复制成 /var/www/html/mailscanner/conf.php

```
shell# cp /var/www/html/mailscanner/conf.php.example /var/www/html/mailscanner/conf.php
```

修改 /var/www/html/mailscanner/conf.php 文件，将 DB_USER 和 DB_PASS

```
define(DB_USER, 'mailwatch');
define(DB_PASS, 'pa5swd');
```

6. 修改 MailScanner 配置

修改 /etc/MailScanner/MailScanner.conf

```
Quarantine Permissions = 0660
Always Looked Up Last = &MailWatchLogging
Is Definitely Not Spam = &SQLWhitelist
Is Definitely Spam = &SQLBlacklist
```

重启动 MailScanner

```
shell# /etc/init.d/MailScanner restart
```

有下面日志内容：

```
Jul 23 16:59:35 red MailScanner[6539]: Config: calling custom init function SQLBlacklist
Jul 23 16:59:35 red MailScanner[6539]: Starting up SQL Blacklist
Jul 23 16:59:35 red MailScanner[6539]: Read 0 blacklist entries
Jul 23 16:59:35 red MailScanner[6539]: Config: calling custom init function
MailWatchLogging
Jul 23 16:59:35 red MailScanner[6539]: Started SQL Logging child
Jul 23 16:59:35 red MailScanner[6539]: Config: calling custom init function SQLWhitelist
Jul 23 16:59:35 red MailScanner[6539]: Starting up SQL Whitelist
```

如果在重启动 MailScanner 的时候，出现下面错误：

```
Shutting down MailScanner daemons:
         MailScanner:       Commmit ineffective while AutoCommit is on at
/usr/lib/MailScanner/MailScanner/CustomFunctions/MailWatch.pm line 97, <CLIENT> line 1.
```

修改 /usr/lib/MailScanner/MailScanner/CustomFunctions/MailWatch.pm 文件，中的 sub ExitLogging 子过程，做如下修改：

```
   $dbh->commit;
```

改成

```
   if (! $dbh->{'AutoCommit'}) {
      $dbh->commit;
   }
```

配置完成了，可以通过浏览器访问访问 MailWatch 的 Web 页面。

5.5.3 测试 MailWatch

1. 发送一封 spamassassin 自带的垃圾邮件样本

```
shell# cd /usr/share/doc/spamassassin-3.2.5/
shell# sendmail test@postfixtest.com <sample-spam.txt
```

查看日志

```
Jul 23 17:09:08 red MailScanner[7197]: Spam Checks: Starting
Jul 23 17:09:08 red MailScanner[7197]: SpamAssassin cache hit for message A0B579C75D.A5BE9
Jul 23 17:09:08 red MailScanner[7197]: Message A0B579C75D.A5BE9 from 127.0.0.1
(root@red.postfix.internal) to postfixtest.com is spam, SpamAssassin (cached,
score=1004.75, required 6, DATE_IN_PAST_96_XX 2.32, DNS_FROM_OPENWHOIS 2.43, GTUBE
1000.00, NO_RELAYS -0.00)
Jul 23 17:09:08 red MailScanner[7197]: Spam Checks: Found 1 spam messages
Jul 23 17:09:08 red MailScanner[7197]: Spam Actions: message A0B579C75D.A5BE9 actions are
store
```

```
Jul 23 17:09:08 red MailScanner[7197]: Spam Checks completed at 41938 bytes per second
Jul 23 17:09:08 red MailScanner[7197]: Deleted 1 messages from processing-database
Jul 23 17:09:08 red MailScanner[7197]: Batch completed at 82 bytes per second (1189 / 14)
Jul 23 17:09:08 red MailScanner[7197]: Batch (1 message) processed in 14.36 seconds
Jul 23 17:09:08 red MailScanner[7197]: Logging message A0B579C75D.A5BE9 to SQL
Jul 23 17:09:08 red MailScanner[7197]: "Always Looked Up Last" took 0.00 seconds
```

2. 发送一封非垃圾邮件样本

```
shell# cd /usr/share/doc/spamassassin-3.2.5/
shell# sendmail test@postfixtest.com <sample-nospam.txt
```

查看日志

```
Jul 23 17:09:38 red MailScanner[7189]: Spam Checks: Starting
Jul 23 17:09:42 red MailScanner[7189]: Message 7ECA19C75D.A8176 from 127.0.0.1
(root@red.postfix.internal) to postfixtest.com is spam, SpamAssassin (not cached,
score=12.298, required 6, autolearn=disabled, DNS_FROM_OPENWHOIS 2.43, URIBL_BLACK 1.96,
URIBL_GREY 0.25, URIBL_OB_SURBL 2.13, URIBL_RHS_DOB 0.90, URIBL_SC_SURBL 2.52,
URIBL_WS_SURBL 2.10)
Jul 23 17:09:42 red MailScanner[7189]: Spam Checks: Found 1 spam messages
Jul 23 17:09:42 red MailScanner[7189]: Spam Actions: message 7ECA19C75D.A8176 actions are
deliver,header
Jul 23 17:09:42 red MailScanner[7189]: Spam Checks completed at 1727 bytes per second
Jul 23 17:09:42 red MailScanner[7189]: Requeue: 7ECA19C75D.A8176 to 4E2789C75C
Jul 23 17:09:42 red MailScanner[7189]: Uninfected: Delivered 1 messages
Jul 23 17:09:42 red MailScanner[7189]: Virus Processing completed at 293329 bytes per
second
Jul 23 17:09:42 red postfix/qmgr[7163]: 4E2789C75C: from=<root@red.postfix.internal>,
size=6731, nrcpt=1 (queue active)
Jul 23 17:09:42 red MailScanner[7189]: Deleted 1 messages from processing-database
Jul 23 17:09:42 red MailScanner[7189]: Batch completed at 385 bytes per second (6961 / 18)
Jul 23 17:09:42 red MailScanner[7189]: Batch (1 message) processed in 18.08 seconds
Jul 23 17:09:42 red MailScanner[7189]: Logging message 7ECA19C75D.A8176 to SQL
Jul 23 17:09:42 red MailScanner[7189]: "Always Looked Up Last" took 0.01 seconds
```

3. 发送一封带附件为可执行文件名的文件（“.bat”，“.exe”，“.com”）的邮件进行测试。

5.5.3 查看 MailWatch Web 页面

1. 邮件状态页面

MailWatch for MailScanner - Recent Messages - Mozilla Firefox

文件(F) 编辑(E) 查看(V) 历史(S) 书签(B) 工具(T) 帮助(H)

http://mail.postfixtest.com/mailscanner/status.php

Google

MailWatch for MailScanner - Rece...

MailWatch
mailwatch.sourceforge.net

Jump to message:

Color Codes
Bad Content/Infected
Spam
High Spam
MCP
High MCP
Whitelisted
Blacklisted
Clean

Status		
MailScanner:	YES	2 children
Postfix:	YES	1 proc(s)
Load Average:	0.17 0.20 0.18	
Mail Queues		
Inbound:		0
Outbound:		0

Today's Totals		
Processed:	6	2.6Mb
Clean:	2	33.3%
Viruses:	0	0.0%
Top Virus:		None
Blocked files:	0	0.0%
Others:	0	0.0%
Spam:	2	33.3%
High Scoring Spam:	1	16.7%
MCP:	0	0.0%
High Scoring MCP:	0	0.0%

Recent Messages | Lists | Quarantine | Reports | Tools/Links | Logout

Last 50 Messages (Refreshing every 30 seconds)							
#	Date/Time	From	To	Subject	Size	SA Score	Status
[ ]	23/07/10 17:19:17		test@postfixtest.com	Warning: E-mail viruses detected	1.3Kb	13.45	Spam
[ ]	23/07/10 17:19:17	postmaster@red.postfix.internal	postmaster	Bad Filename Detected	1.6Kb	5.81	Clean
[ ]	23/07/10 17:19:01	test@postfixtest.com	test2@postfixtest.com	test mail for anti virus	3.3Kb	14.24	Spam Bad Content
[ ]	23/07/10 17:15:43	test@postfixtest.com	test2@postfixtest.com	test mail	2.6Mb	0.00	Clean
[ ]	23/07/10 17:09:42	root@red.postfix.internal	test@postfixtest.com	TBTF ping for 2001-04-20: Reviving	6.8Kb	12.30	Spam
[ ]	23/07/10 17:09:08	root@red.postfix.internal	test@postfixtest.com	Test spam mail (GTUBE)	1.2Kb	1004.75	Spam

Page generated in 4.181308 seconds

完成

2. 邮件隔离区页面

MailWatch for MailScanner - Quarantine Viewer - Mozilla Firefox

文件(F) 编辑(E) 查看(V) 历史(S) 书签(B) 工具(T) 帮助(H)

http://mail.postfixtest.com/mailscanner/quarantine.php?dir=2010072

Google

MailWatch for MailScanner - Quar...

MailWatch
mailwatch.sourceforge.net

Jump to message:

Color Codes
Bad Content/Infected
Spam
High Spam
MCP
High MCP
Whitelisted
Blacklisted
Clean

Status		
MailScanner:	YES	3 children
Postfix:	YES	1 proc(s)
Load Average:	0.33 0.23 0.19	
Mail Queues		
Inbound:		0
Outbound:		0

Today's Totals		
Processed:	6	2.6Mb
Clean:	2	33.3%
Viruses:	0	0.0%
Top Virus:		None
Blocked files:	0	0.0%
Others:	0	0.0%
Spam:	2	33.3%
High Scoring Spam:	1	16.7%
MCP:	0	0.0%
High Scoring MCP:	0	0.0%

Recent Messages | Lists | Quarantine | Reports | Tools/Links | Logout

Folder: 23/07/2010							
#	Date/Time (A/D)	From (A/D)	To (A/D)	Subject (A/D)	Size (A/D)	SA Score (A/D)	Status
[ ]	23/07/10 17:19:01	test@postfixtest.com	test2@postfixtest.com	test mail for anti virus	3.3Kb	14.24	Spam Bad Content
[ ]	23/07/10 17:09:08	root@red.postfix.internal	test@postfixtest.com	Test spam mail (GTUBE)	1.2Kb	1004.75	Spam

Page generated in 5.006268 seconds

完成

5.6 小结

经过以上的配置，病毒和垃圾邮件防护功能已经基本建立，我们还需要调整 MailScanner 的配置文件，定义自己的黑名单，白名单等，可以修改 MailScanner 目录下的相关文件，这里就不再赘述了。

六. 安装Web Mail

Webmail有很多种，如： Squirrelmail，Horde-webmail，Roundcubemail，Extmail，OpenWebmail等。

6.1 准备工作

官方网站：http://roundcube.net/

6.1.1 下载 Roundcubemail

下载地址：http://sourceforge.net/projects/roundcubemail/files/

说明：CentOS 5 系统上一定要用 0.3-stable 版的。CentOS 5 PHP是 5.1.16 版本，Roundcubemail 需求版本是 5.2 所以不要使用 0.3.1 版的 Roundcubemail 。

6.1.2 安装php相关软件

Roundcubemail 是用php编写的一套系统，依赖php的一下包，因此我们把缺少的php包安装上。

```
shell# yum install -y php-xml php-mcrypt
```

6.2 安装 Roundcubemail

6.2.1 安装php相关软件

所谓“安装”也就是把Roundcubemail 软件包解压缩到 WEB 服务器的 DocumentRoot 路径下。

```
shell# tar zxvf roundcubemail-0.3-stable.tar.gz -C /var/www/
shell# cd /var/www/
shell# chown -R vmta.vmta roundcubemail-0.3-stable/
shell# ln -s roundcubemail-0.3-stable/ roundcubemail
```

6.2.2 配置 Mysql 数据库

1. 建立 roundcubemail 数据库和 Mysql 用户

```
mysql> CREATE DATABASE roundcubemail;
mysql> GRANT ALL PRIVILEGES ON roundcubemail.* TO roundcubemail@localhost IDENTIFIED BY
'pa5swd';
```

2. 初始化Mysql 数据库

```
shell# mysql -uroot -p roundcubemail < /var/www/roundcubemail/SQL/mysql.initial.sql
```

6.2.3 配置 Roundcubemail

Roundcubemail 软件包的 config/ 目录下放置的是配置文件，默认只有sample文件，需要自己配置

一下。

1. 配置数据库配置文件

```
shell# cd /var/www/roundcubemail/config
shell# cp db.inc.php.dist db.inc.php
```

修改 /var/www/roundcubemail/config/db.inc.php 文件，做如下修改：

```
$rcmail_config['db_dsnw'] = 'mysql://roundcube:pass@localhost/roundcubemail';
```

改成

```
$rcmail_config['db_dsnw'] = 'mysql://roundcubemail:pa5swd@localhost/roundcubemail';
```

2. 配置主配置文件

```
shell# cd /var/www/roundcubemail/config
shell# cp main.inc.php.dist main.inc.php
```

修改 /var/www/roundcubemail/config/db.inc.php 文件，做如下修改：

修改下面几项内容：

```
$rcmail_config['default_host'] = 'localhost';
$rcmail_config['username_domain'] = 'postfixtest.com';
$rcmail_config['mail_domain'] = 'postfixtest.com';
$rcmail_config['smtp_server'] = 'localhost';
$rcmail_config['product_name'] = 'Webmail';
```

6.3 配置 apache

6.3.1 建立 /etc/httpd/conf.d/roundcubemail.conf 文件

内容如下：

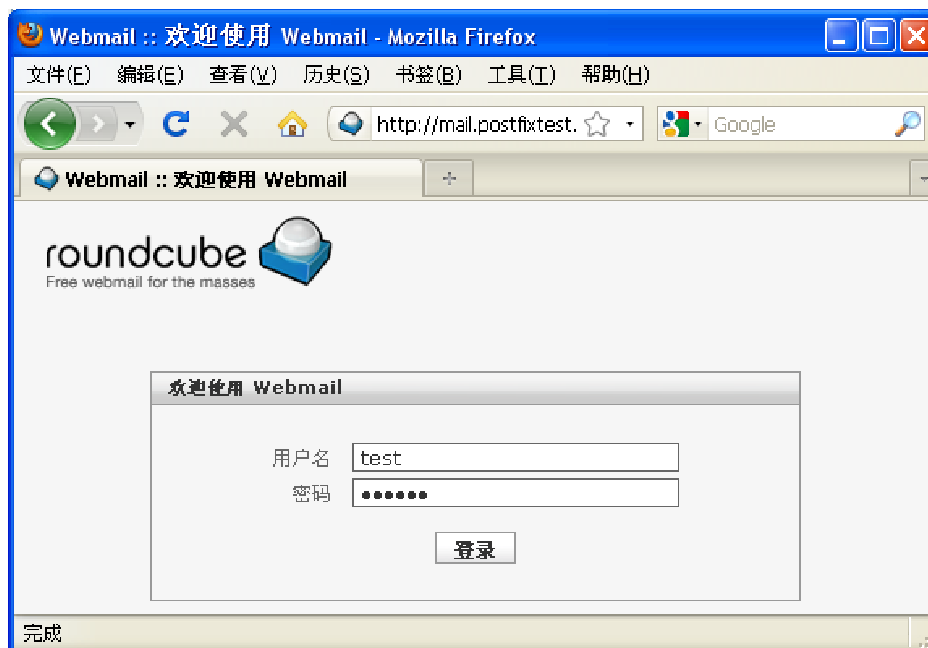
```
Alias /roundcubemail "/var/www/roundcubemail/"
<Directory /var/www/roundcubemail/>
        Options Indexes FollowSymLinks MultiViews
        AllowOverride None
        Order allow,deny
        allow from all
</Directory>
```

6.3.2 重启动 apache

```
shell# /etc/init.d/httpd restart
```

6.3.3 访问 roundcubemail

访问：http://mail.postfixtest.com/roundcubemail



6.4 配置 Roundcubemail 配置插件

6.4.1 修改密码插件

1. 修改配置文件

```
shell# cd /var/www/roundcubemail/plugins/password
shell# cp config.inc.php.dist config.inc.php
```

修改 /var/www/roundcubemail/plugins/password/config.inc.php 文件，需要修改两处。

```
$rcmail_config['password_db_dsn'] = 'mysql://pfadmin:pa5swd@localhost/postfix';
```

```
$rcmail_config['password_query'] = "UPDATE mailbox SET
password=ENCRYPT(%p,concat(_utf8'$1$',right(md5(rand()),8),_utf8'$')) WHERE username=%u
LIMIT 1";
```

2. 制作中文提示文件

建立 zh_CN.inc 文件 ，放到 /var/www/roundcubemail/plugins/password/localization 目录下。
注意：zh_CN.inc 文件必须用 UTF-8 编码保存，否则回乱码。

```
<?php

$labels = array();
$labels['changepasswd']  = '修改密码';
$labels['curpasswd']  = '旧密码:';
```

```
$labels['newpasswd']  = '新密码:';
$labels['confpasswd']  = '新密码确认:';

$messages = array();
$messages['nopassword'] = '请输入新密码。';
$messages['nocurpassword'] = '请输入旧密码。';
$messages['passwordincorrect'] = '旧密码不正确。';
$messages['passwordinconsistency'] = '两次输入的新密码不匹配。';
$messages['crypterror'] = '加密失败，新密码修改失败。';
$messages['connecterror'] = '连接失败，新密码修改失败。';
$messages['internalerror'] = '密码保存失败。';

?>
```

6.4.2 managesieve 插件

managesieve 配置文件基本不用修改，将模板复制成 config.inc.php 即可。

```
shell# cd /var/www/roundcubemail/plugins/managesieve
shell# cp config.inc.php.dist config.inc.php
```

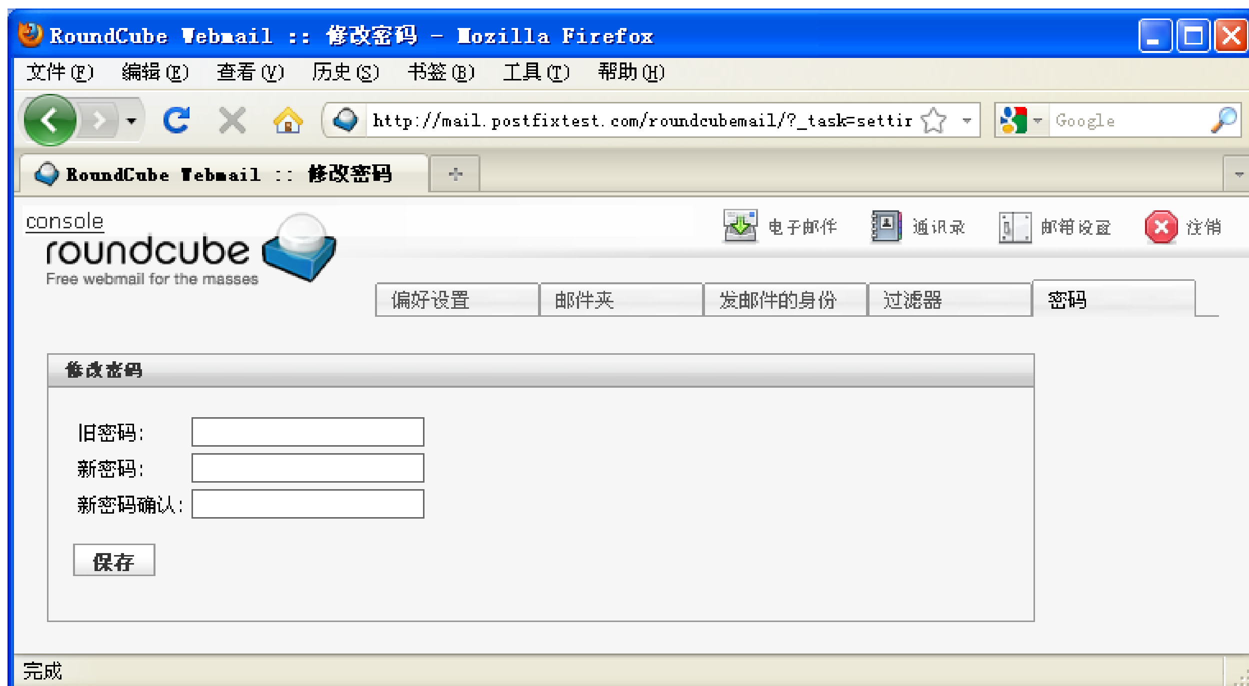
6.4.3 使插件生效

修改 /var/www/roundcubemail/plugins/password/config.inc.php 文件

```
$rcmail_config['plugins'] = array('password','managesieve');
```

6.4.4 测试插件

1. 修改密码界面

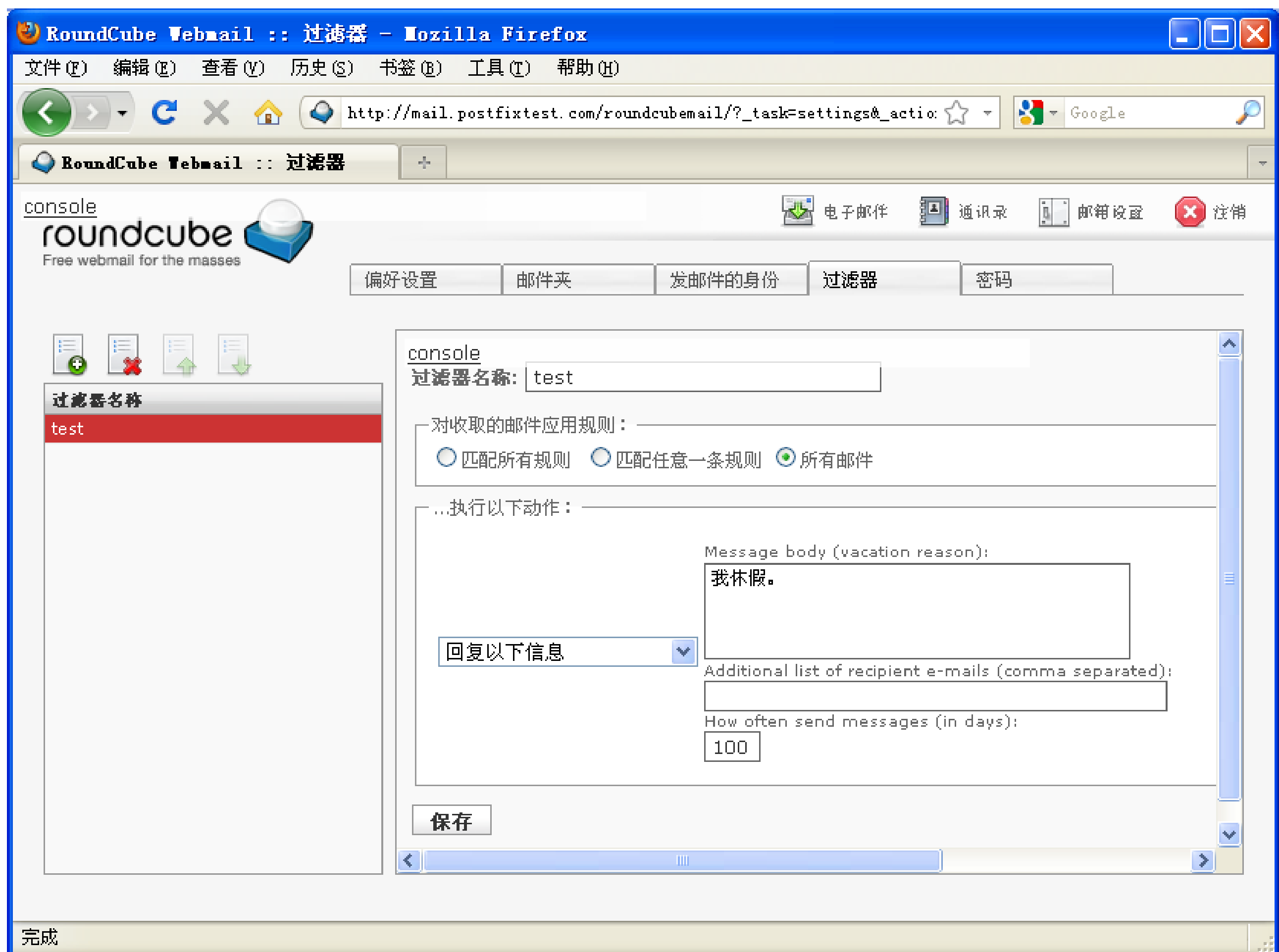


2. managesieve 界面

managesieve 会在用户的目录下建立 sieve 目录，然后 managesieve 插件会在该目录下建立一个名为 roundcube.sieve 的文件，文件里面是 dovecot-sieve 处理邮件的规则。dovecot-sieve 会根据用户的 sieve 目录下的文件文件进行邮件。例如：postfixtest.com 的 test 用户会建立下面文件：

```
postfixtest.com/test/sieve/roundcube.sieve
```

小提示：最好别在服务器端建立过滤规则，我们用 Vacation 功能就行了。一但用户休假了，可以登录到 WEB Mail 设置一下。



七. 配置 SMTP/POP3/HTTP 的TLS功能

7.1 配置 OpenSSL

7.1.1 制作root CA

1. 修改openssl配置文件

修改openssl配置文件：/etc/pki/tls/openssl.cnf

```
dir             = ../../CA              # Where everything is kept
```

改成

```
dir             = /etc/pki/CA           # Where everything is kept
```

2. 安装CA的脚本

```
shell# yum -y install openssl-perl
```

3. 生成Root CA

备份原有文件

```
shell# cd /etc/pki
shell# mv CA CA.bak
```

生成Root CA

```
shell# cd /etc/pki/tls/misc/
shell# ./CA.pl -newca
```

下面为脚本的输出,蓝色为输入部分

```
CA certificate filename (or enter to create)

Making CA certificate ...
Generating a 1024 bit RSA private key
................++++++
.................................++++++
writing new private key to '../../CA/private/cakey.pem'
Enter PEM pass phrase:<YourPassword>
Verifying - Enter PEM pass phrase:<YourPassword>
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
```

```
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [GB]:CN
State or Province Name (full name) [Berkshire]:Liaoning
Locality Name (eg, city) [Newbury]:Dalian
Organization Name (eg, company) [My Company Ltd]:PostfixTest Ltd
Organizational Unit Name (eg, section) []:IT Support
Common Name (eg, your name or your server's hostname) []:mail.postfixtest.com
Email Address []:postmaster@postfixtest.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
Using configuration from /etc/pki/tls/openssl.cnf
Enter pass phrase for ../../CA/private/cakey.pem:<YourPassword>
Check that the request matches the signature
Signature ok
Certificate Details:
        Serial Number:
            bf:22:a5:55:e9:d7:3b:ac
        Validity
            Not Before: Jul 24 02:44:11 2010 GMT
            Not After : Jul 23 02:44:11 2013 GMT
        Subject:
            countryName               = CN
            stateOrProvinceName       = Liaoning
            organizationName          = PostfixTest Ltd
            organizationalUnitName    = IT Support
            commonName                = mail.postfixtest.com
            emailAddress              = postmaster@postfixtest.com
        X509v3 extensions:
            X509v3 Subject Key Identifier:
                91:D0:F2:40:E9:55:CB:66:99:83:28:C8:CE:40:92:FC:6F:89:88:76
            X509v3 Authority Key Identifier:
                keyid:91:D0:F2:40:E9:55:CB:66:99:83:28:C8:CE:40:92:FC:6F:89:88:76
                DirName:/C=CN/ST=Liaoning/O=PostfixTest Ltd/OU=IT
Support/CN=mail.postfixtest.com/emailAddress=postmaster@postfixtest.com
                serial:BF:22:A5:55:E9:D7:3B:AC

            X509v3 Basic Constraints:
                CA:TRUE
Certificate is to be certified until Jul 23 02:44:11 2013 GMT (1095 days)

Write out database with 1 new entries
Data Base Updated
```

生成的文件

私钥文件：/etc/pki/CA/private/cakey.pem
根证书文件： /etc/pki/CA/cacert.pem

修改私钥文件权限

```
shell# chmod 600 /etc/pki/CA/private/cakey.pem
```

7.1.2 生成 req 文件和私钥

1. 建立私钥目录

```
shell# mkdir /etc/pki/myca
shell# cd /etc/pki/myca
```

2. 生成 req 文件和私钥

```
shell# openssl req -new -nodes -keyout mailprivatekey.pem -out mailreq.pem -days 3650
```

下面为输出内容:

```
Generating a 1024 bit RSA private key
...........................++++++
............++++++
writing new private key to 'mailprivatekey.pem'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [GB]:CN
State or Province Name (full name) [Berkshire]:Liaoning
Locality Name (eg, city) [Newbury]:Dalian
Organization Name (eg, company) [My Company Ltd]:PostfixTest Ltd
Organizational Unit Name (eg, section) []:IT Support
Common Name (eg, your name or your server's hostname) []:mail.postfixtest.com
Email Address []:postmaster@postfixtest.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
```

生成的文件

私钥文件：/etc/pki/myca/mailprivatekey.pem
请求文件：/etc/pki/myca/mailreq.pem

修改私钥权限

```
shell# chmod 600
```

查看文件是否生成私钥和 req 文件：

```
shell# ls -l /etc/pki/myca
total 8
-rw------- 1 root root 887 Jul 24 10:49 mailprivatekey.pem
-rw-r--r-- 1 root root 741 Jul 24 10:49 mailreq.pem
```

在上面文件中 mailprivatekey.pem 为私钥 ，mailreq.pem 为 req 文件。

8.1.3 签署 req 文件

```
shell# openssl x509 -req -days 3650 -in mailreq.pem -signkey mailprivatekey.pem -out
mailcert.pem
```

输出内容

```
Signature ok
subject=/C=CN/ST=Liaoning/L=Dalian/O=PostfixTest Ltd/OU=IT
Support/CN=mail.postfixtest.com/emailAddress=postmaster@postfixtest.com
Getting Private key
```

将 root CA 复制到私钥的目录中

```
shell# cp /etc/pki/CA/cacert.pem /etc/pki/myca
```

8.1.4 生成的公钥，私钥文件

```
shell#  ls -l /etc/pki/myca/
total 16
-rw-r--r-- 1 root root 3685 Jul 24 10:58 cacert.pem
-rw-r--r-- 1 root root 1034 Jul 24 10:57 mailcert.pem
-rw------- 1 root root  887 Jul 24 10:49 mailprivatekey.pem
-rw-r--r-- 1 root root  741 Jul 24 10:49 mailreq.pem
```

7.2 配置 Postfix 支持 TLS

7.2.1 配置服务器端 TLS

修改 /etc/postfix/master.cf 文件

增加下面内容：

```
smtps     inet     n     -     n     -     -     smtpd
  -o smtpd_tls_wrappermode=yes -o smtpd_sasl_auth_enable=yes
```

修改 /etc/postfix/main.cf 增加一下几行

```
### tls setting for smtp server
smtpd_tls_key_file  = /etc/pki/myca/mailprivatekey.pem
smtpd_tls_cert_file = /etc/pki/myca/mailcert.pem
smtpd_tls_CAfile    = /etc/pki/myca/cacert.pem
smtpd_use_tls       = yes
smtpd_tls_security_level = may
smtpd_tls_received_header = yes
#smtpd_enforce_tls = yes
smtpd_tls_loglevel = 2
```

说明：默认的，postfix 的 TLS 是不启用的。对于不强制使用 STARTTLS ，使用下面配置：

```
# Postfix 2.3 and later
smtpd_tls_security_level = may
# Obsolete, but still supported
smtpd_use_tls = yes
```

强制要求远程 smtp 使用 STARTTLS ，使用下面配置：

```
# Postfix 2.3 and later
smtpd_tls_security_level = encrypt
# Obsolete, but still supported
smtpd_enforce_tls = yes
```

7.2.2 配置客户器端 TLS

```
### tls setting for smtp client
smtp_use_tls       = yes
smtp_tls_key_file  = /etc/pki/myca/mailprivatekey.pem
smtp_tls_cert_file = /etc/pki/myca/mailcert.pem
smtp_tls_CAfile    = /etc/pki/myca/cacert.pem
#smtp_tls_policy_maps = hash:/etc/postfix/tls_policy_maps
```

重启动 postfix 服务

7.3 配置 Dovecot 支持 TLS

在 /etc/dovecot.conf 中添加如下内容：

```
verbose_ssl = no
ssl_key_file = /etc/pki/myca/mailprivatekey.pem
ssl_cert_file = /etc/pki/myca/mailcert.pem
```

重启动 dovecot 服务

7.4 配置 Apache 支持 TLS

7.4.1 配置 apache 的 ssl 功能

在 /etc/httpd/conf.d/ssl.conf 中添加如下内容：

```
SSLCertificateFile /etc/pki/tls/certs/localhost.crt

SSLCertificateKeyFile /etc/pki/tls/private/localhost.key
```

改成：

```
SSLCertificateFile /etc/pki/myca/mailcert.pem

SSLCertificateKeyFile /etc/pki/myca/mailprivatekey.pem
```

```
ServerName mail.postfixtest.com
```

重启动 apache 服务

7.5 收发邮件 TLS

7.5.1 发邮件测试

这个例子中用 postfixtest.com 域给 qmailtest.com 域发邮件，查看两个 MTA 之间发送邮件的时候是否启用 TLS 方式通信，需要检查 2 个地方：

1. 查看 postfix 的日志，看发送邮件时候有没有 TLS 的信息；
2. 查看 qmailtest.com 域用户的信头信息，看有没有 TLS 的信息。

发信时候，在客户端使用 TLS 发信，检查内容：

1. 给其他 MTA 发送邮件的日志

```
Jul 24 11:52:56 postfix postfix/smtpd[2877]: connect from unknown[192.168.1.100]
Jul 24 11:52:56 postfix postfix/smtpd[2877]: setting up TLS connection from
unknown[192.168.1.100]
Jul 24 11:52:56 postfix postfix/smtpd[2877]: Anonymous TLS connection established from
unknown[192.168.1.100]: TLSv1 with cipher RC4-MD5 (128/128 bits)
Jul 24 11:52:56 postfix postfix/smtpd[2877]: CE57594959: client=unknown[192.168.1.100],
sasl_method=LOGIN, sasl_username=test@postfixtest.com
Jul 24 11:52:56 postfix postfix/cleanup[2885]: CE57594959: message-
id=<DBE5E5305933443E9FE82B302C5D8796@zhoulj>
Jul 24 11:52:56 postfix postfix/qmgr[2697]: CE57594959: from=<test@postfixtest.com>,
size=1510, nrcpt=1 (queue active)
Jul 24 11:52:56 postfix postfix/smtpd[2877]: disconnect from unknown[192.168.1.100]
Jul 24 11:52:57 postfix postfix/smtp[2887]: certificate verification failed for
mx1.qmailtest.com[192.168.1.80]:25: untrusted issuer /O=Qmail Toaster Server/OU=For
testing purposes only/CN=www.qmailtoaster.com
Jul 24 11:52:57 postfix postfix/smtp[2887]: CE57594959: to=<test@qmailtest.com>,
relay=mx1.qmailtest.com[192.168.1.80]:25, delay=0.34, delays=0.08/0.02/0.22/0.02,
dsn=2.0.0, status=sent (250 ok 1279943578 qp 6525)
Jul 24 11:52:57 postfix postfix/qmgr[2697]: CE57594959: removed
```

2. 查看 qmailtest.com 域用户收到的信头信息

```
Return-Path: <test@postfixtest.com>
Delivered-To: test@qmailtest.com
Received: (qmail 6526 invoked by uid 89); 24 Jul 2010 03:52:58 -0000
Received: from unknown (HELO red.postfix.internal) (192.168.1.70)
     by blue.qmail.internal with (DHE-RSA-AES256-SHA encrypted) SMTP; 24 Jul 2010 03:52:58
-0000
Received-SPF: pass (blue.qmail.internal: SPF record at postfixtest.com designates
192.168.1.70 as permitted sender)
Received: from creditzhoulj (unknown [192.168.1.100])
     (using TLSv1 with cipher RC4-MD5 (128/128 bits))
     (No client certificate requested)
     by red.postfix.internal (Postfix) with ESMTPSA id CE57594959
     for <test@qmailtest.com>; Sat, 24 Jul 2010 11:52:56 +0800 (CST)
Message-ID: <DBE5E5305933443E9FE82B302C5D8796@creditzhoulj>
From: <test@postfixtest.com>
To: <test@qmailtest.com>
Subject: test mail
```

7.5.2 从其他 MTA 接收邮件测试

这个例子中用 qmailtest.com 域给 postfixtest.com 域发邮件，查看两个 MTA 之间发送邮件的时候是否启用 TLS 方式通信，需要检查两个地方：

1. 查看 postfix 的日志，看接收邮件的时候有没有 TLS 的信息；
2. 查看 postfixtest.com 域用户的信头信息，看有没有 TLS 的信息。

检查内容：

1. 接收其他域邮件的日志

```
Jul 24 11:59:07 postfix postfix/smtpd[2890]: connect from unknown[192.168.1.80]
Jul 24 11:59:08 postfix postfix/smtpd[2890]: setting up TLS connection from
unknown[192.168.1.80]
Jul 24 11:59:08 postfix postfix/smtpd[2890]: Anonymous TLS connection established from
unknown[192.168.1.80]: TLSv1 with cipher DHE-RSA-AES256-SHA (256/256 bits)
Jul 24 11:59:08 postfix postfix/smtpd[2890]: 1186F94959: client=unknown[192.168.1.80]
Jul 24 11:59:08 postfix postfix/cleanup[2897]: 1186F94959: message-
id=<eaac39ac17db174b65d61714bf399276.squirrel@192.168.1.80>
Jul 24 11:59:08 postfix postfix/qmgr[2697]: 1186F94959: from=<test@qmailtest.com>,
size=1054, nrcpt=1 (queue active)
Jul 24 11:59:08 postfix postfix/smtpd[2890]: disconnect from unknown[192.168.1.80]
Jul 24 11:59:08 postfix postfix/pipe[2899]: 1186F94959: to=<test@postfixtest.com>,
relay=dovecot, delay=0.18, delays=0.14/0.01/0/0.03, dsn=2.0.0, status=sent (delivered via
dovecot service)
Jul 24 11:59:08 postfix postfix/qmgr[2697]: 1186F94959: removed
```

2. 查看 postfixtest.com 域用户收到的信头信息

```
Return-Path: <test@qmailtest.com>
Delivered-To: test@postfixtest.com
```

```
Received: from blue.qmail.internal (unknown [192.168.1.80])
    (using TLSv1 with cipher DHE-RSA-AES256-SHA (256/256 bits))
    (No client certificate requested)
    by red.postfix.internal (Postfix) with ESMTPS id 1186F94959
    for <test@postfixtest.com>; Sat, 24 Jul 2010 11:59:08 +0800 (CST)
Received: (qmail 6602 invoked by uid 89); 24 Jul 2010 03:59:08 -0000
Received: from unknown (HELO ?192.168.1.80?) (127.0.0.1)
  by blue.qmail.internal with SMTP; 24 Jul 2010 03:59:08 -0000
Received: from 192.168.1.100
        (SquirrelMail authenticated user test@qmailtest.com)
        by 192.168.1.80 with HTTP;
        Sat, 24 Jul 2010 11:59:08 +0800
Message-ID: <eaac39ac17db174b65d61714bf399276.squirrel@192.168.1.80>
Date: Sat, 24 Jul 2010 11:59:08 +0800
Subject: test mail from qmailtest
From: test@qmailtest.com
To: test@postfixtest.com
```

7.5.3 查看 Dovecot 日志

在邮件客户中选择使用 TLS 收信， 查看 dovecot 日志，有如下内容：

```
Jul 24 12:13:40 postfix dovecot: pop3-login: Disconnected (auth failed, 5 attempts):
user=<test@test.com>, method=LOGIN, rip=192.168.1.100, lip=192.168.1.70, TLS
Jul 24 12:13:49 postfix dovecot: pop3-login: Login: user=<test@postfixtest.com>,
method=PLAIN, rip=192.168.1.100, lip=192.168.1.70, TLS
Jul 24 12:13:50 postfix dovecot: POP3(test@postfixtest.com): Disconnected: Logged out
top=0/0, retr=11/10375, del=11/11, size=10194
```

7.5.4 查看 Apache 日志

访问：https://mail.postfixtest.com/roundcubemail ，查看 /var/log/httpd/ssl_request_log 日志，有如下内容：

```
[24/Jul/2010:12:21:06 +0800] 192.168.1.100 TLSv1 DHE-RSA-AES256-SHA "GET
/roundcubemail/skins/default/images/display/loading.gif HTTP/1.1" 2942
[24/Jul/2010:12:21:05 +0800] 192.168.1.100 TLSv1 DHE-RSA-AES256-SHA "GET /roundcubemail/?
_task=mail&_action=list&_mbox=INBOX&_page=1&_remote=1&_=1280891631048&_unlock=1 HTTP/1.1"
413
```

八. 日志管理

8.1 安装mailgraph

mailgraph 是一个非常简单的使用perl编写的基于RRDtool 的 postfix 和 sendmail 的日志分析工具，通过分析日志生产天、周、月、年的图片报表。

8.1.1 安装 mailgraph 依赖的 perl 模块和 RRDtool 模块

1. 安装 RRDtool

官方网站：http://oss.oetiker.ch/rrdtool/
下载地址：http://packages.sw.be/rrdtool/

mailgraph 是以 RRDtool 为基础开发的，所以先安装 RRDtool 模块：

```
shell# rpm -ivh rrdtool-1.2.30-1.el5.rf.i386.rpm perl-rrdtool-1.2.30-1.el5.rf.i386.rpm
```

2. 安装 perl-File-Tail包

下载地址：http://packages.sw.be/perl-File-Tail/

```
shell# rpm -ivh perl-File-Tail-0.99.3-1.2.el5.rf.noarch.rpm
```

8.1.2 安装 mailgraph

mailgraph是David Schweikert 编写的。
下载地址：http://mailgraph.schweikert.ch/

```
shell# tar zxvf mailgraph-1.14.tar.gz -C /usr/local/
shell# cd /usr/local/
shell# ln -s mailgraph-1.14/ mailgraph
shell# cd mailgraph
shell# mkdir /var/www/mailgraph
shell# cp mailgraph.cgi mailgraph.css /var/www/mailgraph
shell# chown -R vmta /var/www/mailgraph
```

修改 mailgraph.pl 指定 .rrd 文件路径

```
my $rrd = "mailgraph.rrd";
my $rrd_virus = "mailgraph_virus.rrd";
```

改成

```
my $rrd = "/var/www/mailgraph/mailgraph.rrd";
my $rrd_virus = "/var/www/mailgraph/mailgraph_virus.rrd";
```

修改mailgraph-init

```
MAILGRAPH_PL=/usr/local/bin/mailgraph.pl
MAIL_LOG=/var/log/syslog
```

改成

```
MAILGRAPH_PL=/usr/local/mailgraph/mailgraph.pl
MAIL_LOG=/var/log/maillog
```

配置 mailgraph 启动脚本

```
shell# cp mailgraph-init /etc/init.d/mailgraph
shell# chmod 755 /etc/init.d/mailgraph
shell# /etc/init.d/mailgraph start
shell# chkconfig mailgraph on
```

配置 apache ，创建 /etc/httpd/conf.d/mailgraph.conf

```
shell# vi /etc/httpd/conf.d/mailgraph.conf
```

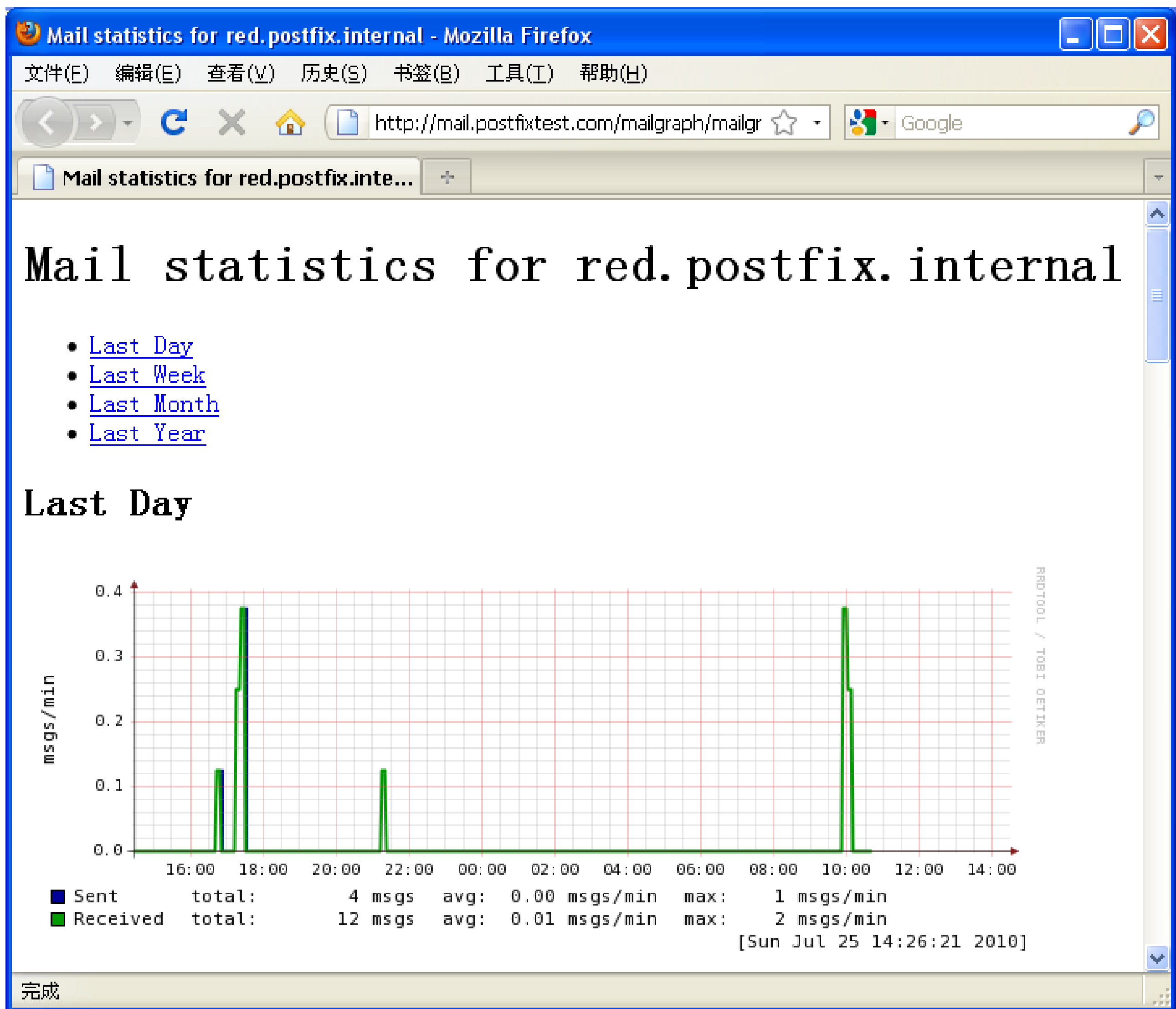
增加如下内容：

```
ScriptAlias /mailgraph/ "/var/www/mailgraph/"
<Directory /var/www/mailgraph/>
   AllowOverride AuthConfig
   AuthType Basic
   AuthName "Restricted Files"
   AuthUserFile /etc/httpd/conf/mailadmin.passwd
   Require user mailadmin
   Options ExecCGI
   Order deny,allow
   Deny from all
   Allow from 127.0.0.1 192.168.1
</Directory>
```

重启动 apache

```
shell# /etc/init.d/httpd reload
```

通过浏览器访问 mailgraph 页面，地址：http://mail.postfixtest.com/mailgraph/mailgraph.cgi



8.2 安装 awstats

Awstats 是非常优秀的日志分析工具，可以分析 Web,Mail,FTP 等日志。

官方网址： http://awstats.sourceforge.net/

8.2.1 安装 awstats

```
shell# tar zxvf awstats-6.95.tar.gz -C /usr/local/
shell# cd /usr/local/
shell# ln -s awstats-6.95/ awstats
```

Awstats 目录下 docs 目录里面有详细说文档。Mail 日志分析的文档为 awstats_faq.html 文件，查找 FAQ-COM100，是介绍 mail 日志相关设置的。

8.2.2 使用 awstats 分析 Web 日志

1. 生成配置文件

```
shell# cd /usr/local/awstats/tools
shell# ./awstats_configure.pl
```

按照提示输入下面信息：
输入 apache 的配置文件位置：/etc/httpd/conf/httpd.conf

```
Config file path ('none' to skip web server setup):
> /etc/httpd/conf/httpd.conf
```

创建一个新的 AWStats 配置文件

```
-----> Need to create a new config file ?
Do you want me to build a new AWStats config/profile
file (required if first install) [y/N] ? y
```

输入 profile 名称：web

```
Your web site, virtual server or profile name:
> web
```

输入 awstats 配置文件位置：

```
-----> Define config file path
In which directory do you plan to store your config file(s) ?
Default: /etc/awstats
Directory path to store config file(s) (Enter for default):
>
```

记住 url 地址：http://localhost/awstats/awstats.pl?config=web

2. 建立需要的目录

```
shell# mkdir -p /usr/local/awstats/data/web
```

3. 修改 awstats 配置 /etc/awstats/awstats.web.conf

```
LogFile="/var/log/httpd/access_log"

DirData="/usr/local/awstats/data/web"
```

4. 分析日志

```
shell# perl /usr/local/awstats/wwwroot/cgi-bin/awstats.pl -config=web -update
```

5. 访问 web 页面

http://mail.postfixtest.com/awstats/awstats.pl?config=web



8.2.3 使用 awstats 分析 Mail 日志

1. 生成配置文件

```
shell# cd /usr/local/awstats/tools
shell# ./awstats_configure.pl
```

按照提示输入下面信息：
输入 apache 的配置文件位置：/etc/httpd/conf/httpd.conf

```
Config file path ('none' to skip web server setup):
> /etc/httpd/conf/httpd.conf
```

创建一个新的 AWStats 配置文件

```
-----> Need to create a new config file ?
Do you want me to build a new AWStats config/profile
file (required if first install) [y/N] ? y
```

输入 profile 名称：mail

```
Your web site, virtual server or profile name:
> mail
```

输入 awstats 配置文件位置：

```
-----> Define config file path
In which directory do you plan to store your config file(s) ?
Default: /etc/awstats
Directory path to store config file(s) (Enter for default):
>
```

记住 url 地址：http://localhost/awstats/awstats.pl?config=mail

2. 建立需要的目录

```
shell# mkdir -p /usr/local/awstats/data/mail
```

3. 改 awstats 配置 /etc/awstats/awstats.mail.conf

```
LogFile="perl /usr/local/awstats/tools/maillogconvert.pl standard < /var/log/maillog |"

LogType=M

LogFormat="%time2 %email %email_r %host %host_r %method %url %code %bytesd"

DirData="/usr/local/awstats/data/mail"
```

4. 分析日志

```
shell# perl /usr/local/awstats/wwwroot/cgi-bin/awstats.pl -config=mail -update
```

5. 访问 web 页面

http://mail.postfixtest.com/awstats/awstats.pl?config=mail



8.2.3 自动分析日志设置

crontab 中加入如下内容

```
0 */4 * * * /usr/bin/perl /usr/local/awstats/wwwroot/cgi-bin/awstats.pl -config=mail
-update
0 */4 * * * /usr/bin/perl /usr/local/awstats/wwwroot/cgi-bin/awstats.pl -config=web
-update
```

九. 备份系统

9.1 备份系统文件

/etc/hosts
/etc/sysconfig/network
/etc/sysconfig/network-scripts/ifcfg-eth0

9.2 邮件系统配置

9.2.1 Postfix 配置文件

/etc/postfix/main.cf
/etc/postfix/master.cf
/etc/postfix/mysql_virtual_alias_maps.cf
/etc/postfix/mysql_virtual_domains_maps.cf
/etc/postfix/mysql_virtual_mailbox_maps.cf
/etc/postfix/mysql_virtual_sender_maps.cf

9.2.2 dovecot 配置

/etc/dovecot.conf
/etc/dovecot-mysql.conf

9.2.3 MailScanner 配置文件

/etc/MailScanner/目录

9.2.4 apache 配置文件

/etc/httpd/conf/httpd.conf
/etc/httpd/conf/mailadmin.passwd
/etc/httpd/conf.d/mailgraph.conf
/etc/httpd/conf.d/mailwatch.conf
/etc/httpd/conf.d/postfixadmin.conf
/etc/httpd/conf.d/roundcubemail.conf
/etc/httpd/conf.d/ssl.conf

9.2.5 awstats 配置文件

/etc/awstats/awstats.mail.conf
/etc/awstats/awstats.web.conf

9.2.6 证书文件
/etc/pki/myca/

9.3 数据备份

9.3.1 Mysql 数据库备份

postfix
mailscanner

```
mysqldump -uroot -p postfix > postfixadmin_mysql.sql
mysqldump -uroot -p mailscanner > mailscanner_mysql.sql
```

9.3.2 应用程序

/usr/local/awstats-6.95
/usr/local/mailgraph-1.14
/usr/local/mailwatch-1.0.5
/var/www/mailgraph
/var/www/postfixadmin-2.3
/var/www/roundcubemail-0.3-stable
/var/www/html/mailscanner

9.3.3 脚本

/usr/local/bin/maildir-creation.sh
/usr/local/bin/maildir-deletion.sh

9.3.4 邮件目录

/var/vmta/

参考文献

文档：

1. 《Postfix 权威指南》
2. 《apache 经典实例》
3. 《Bind and DNS》

网站：

http://www.bowe.id.au/michael/isp/postfix-server.htm
http://workaround.org/ispmail/lenny

官方文档：

http://www.postfix.org/documentation.html
http://wiki.dovecot.org
http://wiki.apache.org/spamassassin/FrontPage
http://www.mailscanner.info/documentation.html
http://mailwatch.sourceforge.net/doku.php